

Magnetic Resonance Spectroscopy and Imaging: NMR, MRI and ESR

Contents

Articles

Nuclear Magnetic Resonance	1
Version 3.0	2
Relaxation	2
Chemical Shift	3
Knight shift	7
Robinson oscillator	8
Fourier transform	8
Discrete Fourier transform	30
Fast Fourier transform	43
Fourier transform spectroscopy	51
Nuclear magnetic resonance	55
NMR Spectroscopy	70
2D-NMR	76
User:Bci2/2D-FT NMRI and Spectroscopy	78
Solid-state nuclear magnetic resonance	84
Magnetic resonance microscopy	90
Imaging	92
Medical imaging	92
MRI	99
ESR Spectroscopy and Microspectroscopy	118
ESR	118
References	
Article Sources and Contributors	127
Image Sources, Licenses and Contributors	129
Article Licenses	
License	130

Nuclear Magnetic Resonance

Version 3.0

Relaxation

Relaxation stands quite generally for a release of tension, a return to equilibrium.

In the sciences, the term is used in the following ways:

- Relaxation (physics), and more in particular:
 - Relaxation (NMR), processes by which nuclear magnetization returns to the equilibrium distribution
 - Dielectric relaxation, the delay in the dielectric constant of a material
 - Structural relaxation, responsible for the glass transition
- In mathematics:
 - Relaxation technique (mathematics), a technique for transforming hard constraints into easier ones
 - Relaxation method, for numerically solving elliptic partial differential equations
- In computer science:
 - Relaxation, the act of substituting alternative program code during linking

In Physiology, Hypnotism, Meditation, Recreation:

- Relaxation technique, an activity that helps a person to relax
- Relaxed in Flow (psychology), a state of arousal, flow, over-learned self-control and relaxation
- Relaxation (psychology), the emotional state of low tension

See also:

- Tension (music)
-

Chemical Shift

In nuclear magnetic resonance (NMR), the **chemical shift** describes the dependence of nuclear magnetic energy levels on the electronic environment in a molecule.^{[1] [2] [3]} Chemical shifts are relevant in NMR spectroscopy techniques such as proton NMR and carbon-13 NMR.

An atomic nucleus can have a magnetic moment (nuclear spin), which gives rise to different energy levels and resonance frequencies in a magnetic field. The total magnetic field experienced by a nucleus includes local magnetic fields induced by currents of electrons in the molecular orbitals (note that electrons have a magnetic moment themselves). The electron distribution of the same type of nucleus (e.g. ^1H , ^{13}C , ^{15}N) usually varies according to the local geometry (binding partners, bond lengths, angles between bonds, ...), and with it the local magnetic field at each nucleus. This is reflected in the spin energy levels (and resonance frequencies). The variations of nuclear magnetic resonance frequencies of the same kind of nucleus, due to variations in the electron distribution, is called the chemical shift. The size of the chemical shift is given with respect to a reference frequency or reference sample (see also *chemical shift referencing*), usually a molecule with a barely distorted electron distribution.

The chemical shift is of great importance for NMR spectroscopy, a technique to explore molecular properties by looking at nuclear magnetic resonance phenomena.

Operating frequency

The operating frequency ω_0 of a magnet is calculated from the Larmor equation

$$\omega_0 = \gamma * B_0$$

where B_0 is the actual strength of the magnet in units like teslas or gauss, and γ is the gyromagnetic ratio of the nucleus being tested which is in turn calculated from its magnetic moment μ and spin number I with the nuclear magneton μ_N and the Planck constant h :

$$\gamma = \frac{\mu \mu_N}{hI}$$

Thus, the proton operating frequency for a 1 T magnet is calculated as:

$$\omega_0 = \gamma B_0 = \frac{2.79 \times 5.05 \times 10^{-27} \text{ J/T}}{6.62 \times 10^{-34} \text{ Js} \times (1/2)} \times 1 \text{ T} = 42.5 \text{ MHz}$$

Chemical shift referencing

Chemical shift δ is usually expressed in parts per million (ppm) by frequency, because it is calculated from:

$$\delta = \frac{\text{difference in precession frequency between two nuclei}}{\text{operating frequency of the magnet}}$$

Since the numerator is usually in hertz, and the denominator in megahertz, delta is expressed in ppm.

The detected frequencies (in Hz) for ^1H , ^{13}C , and ^{29}Si nuclei are usually referenced against TMS (tetramethylsilane) or DSS, which is assigned the chemical shift of zero. Other standard materials are used for setting the chemical shift for other nuclei.

Thus, an NMR signal at 300 Hz from TMS at an applied frequency of 300 MHz has a chemical shift of:

$$\frac{300 \text{ Hz}}{300 \times 10^6 \text{ Hz}} = 1 \times 10^{-6} = 1 \text{ ppm}$$

Although the frequency depends on the applied field the chemical shift is independent of it. On the other hand the resolution of NMR will increase with applied magnetic field resulting in ever increasing chemical shift changes.

The induced magnetic field

The electrons around a nucleus will circulate in a magnetic field and create a secondary induced magnetic field. This field opposes the applied field as stipulated by Lenz's law and atoms with higher induced fields (i.e., higher electron density) are therefore called *shielded*, relative to those with lower electron density. The chemical milieu of an atom can influence its electron density through the polar effect. Electron-donating alkyl groups, for example, lead to increased shielding while electron-withdrawing substituents such as nitro groups lead to *deshielding* of the nucleus. Not only substituents cause local induced fields. Bonding electrons can also lead to shielding and deshielding effects. A striking example of this are the pi bonds in benzene. Circular current through the hyperconjugated system causes a shielding effect at the molecule's center and a deshielding effect at its edges. Trends in chemical shift are explained based on the degree of shielding or deshielding.

Nuclei are found to resonate in a wide range to the left (or more rare to the right) of the internal standard. When a signal is found with a higher chemical shift:

- the applied effective magnetic field is lower, if the resonance frequency is fixed, (as in old traditional CW spectrometers)
- the frequency is higher, when the applied magnetic field is static, (normal case in FT spectrometers)
- the nucleus is more deshielded
- the signal or shift is **downfield** or at **low field** or paramagnetic

Conversely a lower chemical shift is called a **diamagnetic shift**, and is **upfield** and more shielded.

Diamagnetic shielding

In real molecules protons are surrounded by a cloud of charge due to adjacent bonds and atoms. In an applied magnetic field (B_0) electrons circulate and produce an induced field (B_1) which opposes the applied field. The effective field at the nucleus will be $B = B_0 - B_1$. The nucleus is said to be experiencing a diamagnetic shielding

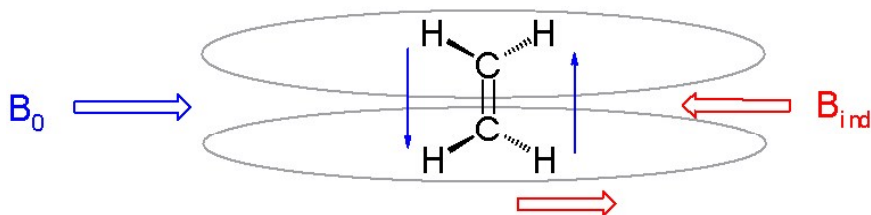
Factors causing chemical shifts

Important factors influencing chemical shift are electron density, electronegativity of neighboring groups and anisotropic induced magnetic field effects.

Electron density shields a nucleus from the external field. For example in proton NMR the electron-poor tropylium ion has its protons downfield at 9.17 ppm, those of the electron-rich cyclooctatetraenyl anion move upfield to 6.75 ppm and its dianion even more upfield to 5.56 ppm.

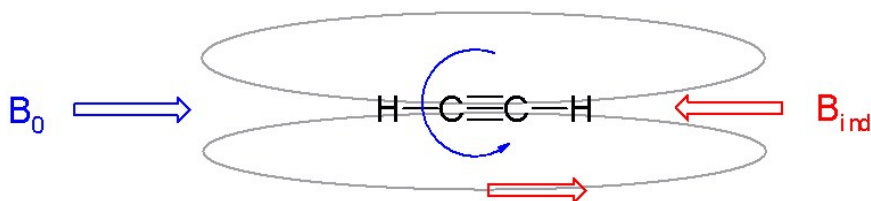
A nucleus in the vicinity of an electronegative atom experiences reduced electron density and the nucleus is therefore deshielded. In proton NMR of methyl halides (CH_3X) the chemical shift of the methyl protons increase in the order $I < Br < Cl < F$ from 2.16 ppm to 4.26 ppm reflecting this trend. In carbon NMR the chemical shift of the carbon nuclei increase in the same order from around -10 ppm to 70 ppm. Also when the electronegative atom is removed further away the effect diminishes until it can be observed no longer.

Anisotropic induced magnetic field effects are the result of a local induced magnetic field experienced by a nucleus resulting from circulating electrons that can either be paramagnetic when it is parallel to the applied field or diamagnetic when it is opposed to it. It is observed in alkenes where the double bond is oriented perpendicular to the external field with pi electrons likewise circulating at right angles. The induced magnetic field lines are parallel to the external field at the location of the alkene protons which therefore shift downfield to a 4.5 ppm to 7.5 ppm range. The three-dimensional space where a nucleus experiences diamagnetic shift is called the shielding zone with a cone-like shape aligned with the external field.



The protons in aromatic compounds are shifted downfield even further with a signal for benzene at 7.73 ppm as a consequence of a diamagnetic ring current.

Alkyne protons by contrast resonate at high field in a 2–3 ppm range. For alkynes the most effective orientation is the external field in parallel with electrons circulation around the triple bond. In this way the acetylenic protons are located in the cone-shaped shielding zone hence the upfield shift.



Magnetic properties of most common nuclei

^1H and ^{13}C aren't the only nuclei susceptible to NMR experiments. A number of different nuclei can also be detected, although the use of such techniques is generally rare due to small relative sensitivities in NMR experiments (compared to ^1H) of the nuclei in question, the other factor for rare use being their slender representation in nature/organic compounds.

Isotope	Occurrence in nature (%)	spin number I	Magnetic moment $\mu^{[4]}$	Electric quadrupole moment ($\times 10^{-24} \text{ cm}^2$)	Operating frequency at 7 T (MHz)	Relative sensitivity
^1H	99.984	1/2	2.79628		300.13	1
^2H	0.016	1	0.85739	2.8×10^{-3}	46.07	0.0964
^{10}B	18.8	3	1.8005	7.4×10^{-2}	32.25	0.0199
^{11}B	81.2	3/2	2.6880	2.6×10^{-2}	96.29	0.165
^{12}C	98.9	0				
^{13}C	1.1	1/2	0.70220		75.47	0.0159
^{14}N	99.64	1	0.40358	7.1×10^{-2}	21.68	0.00101
^{15}N	0.37	1/2	-0.28304		30.41	0.00104
^{16}O	99.76	0				
^{17}O	0.0317	5/2	-1.8930	-4.0×10^{-3}	40.69	0.0291
^{19}F	100	1/2	2.6273		282.40	0.834
^{28}Si	92.28	0				
^{29}Si	4.70	1/2	-0.55548		59.63	0.0785

^{31}P	100	1/2	1.1205		121.49	0.0664
^{35}Cl	75.4	3/2	0.92091	-7.9×10^{-2}	29.41	0.0047
^{37}Cl	24.6	3/2	0.68330	-6.2×10^{-2}	24.48	0.0027
Magnetic properties of common nuclei ^[5]						

^1H , ^{13}C , ^{15}N , ^{19}F and ^{31}P are the five nuclei that have the greatest importance in NMR experiments:

- ^1H because of high sensitivity and vast occurrence in organic compounds
- ^{13}C because of being the key component of all organic compounds despite occurring at a low abundance (1.1%) compared to the major isotope of carbon ^{12}C , which has a spin of 0 and therefore is NMR inactive.
- ^{15}N because of being a key component of important biomolecules such as proteins and DNA
- ^{19}F because of high relative sensitivity
- ^{31}P because of frequent occurrence in organic compounds and moderate relative sensitivity

Other chemical shifts

The related Knight shift (first reported in 1949) is observed with pure metals. The NMR chemical shift in its present day meaning first appeared in journals in 1950. Chemical shifts with a different meaning appear in X-ray photoelectron spectroscopy as the shift in atomic core-level energy due to a specific chemical environment. The term is also used in Mössbauer spectroscopy, where similarly to NMR it refers to a shift in peak position due to the local chemical bonding environment. As is the case for NMR the chemical shift reflects the electron density at the atomic nucleus.^[6]

See also

- Carbon-13 NMR
- MRI
- NMR spectroscopy
- 2D-FT NMRI and Spectroscopy
- Nuclear magnetic resonance
- Protein NMR
- Proton NMR
- Solid-state NMR
- Zeeman effect

External links

- www.chem.wisc.edu^[7]
- [BioMagResBank](#)^[8]
- wwwchem.csustan.edu^[9]
- [Proton chemical shifts](#)^[7]
- [Carbon chemical shifts](#)^[10]
- Online tutorials (these generally involve combined use of IR, ^1H NMR, ^{13}C NMR and mass spectrometry)
 - Problem set 1, advanced^[11] (see also this link^[12] for more background information on spin-spin coupling)
 - Problem set 2, moderate^[13]
 - Problem set 4, moderate, German language (don't let that scare you away!)^[14]
 - Problem set 5, the best!^[15]

- Combined solutions to problem set 5 (Problems 1–32) ^[16] and (Problems 33–64) ^[17]

References

- [1] *Spectrometric Identification of organic Compounds* Silverstein, Bassler, Morrill 4th Ed. ISBN 047109700
- [2] *Organic Spectroscopy* William Kemp 3rd Ed. ISBN 0333417674
- [3] *Basic ¹H - ¹³C-NMR spectroscopy* Metin Balei ISBN 0444518118
- [4] In units of the nuclear magneton
- [5] CRC Handbook of Chemistry and Physics 65Th Ed
- [6] *A Short History of Three Chemical Shifts* Shin-ichi Nagaoka Vol. 84 No. 5 May **2007** Journal of Chemical Education 801
- [7] <http://www.chem.wisc.edu/areas/reich/handouts/nmr-h/hdata.htm>
- [8] <http://www.bmrwisc.edu>
- [9] <http://wwwchem.csustan.edu/Tutorials/NMRTABLE.HTM>
- [10] <http://www.chem.wisc.edu/areas/reich/handouts/nmr-c13/cdata.htm>
- [11] <http://www.chem.ucla.edu/~webspectra/>
- [12] <http://drx.ch.huji.ac.il/nmr/whatisnmr/whatisnmr.html>
- [13] <http://orgchem.colorado.edu/hndbksupport/spectprob/problems.html>
- [14] <http://www.chem.uni-potsdam.de/tools/kombi1.htm>
- [15] <http://www.nd.edu/~smithgrp/structure/workbook.html>
- [16] <http://www.nd.edu/~smithgrp/structure/answers1-32.GIF>
- [17] <http://www.nd.edu/~smithgrp/structure/answers33-64.GIF>

Knight shift

The **Knight shift** is a shift in the nuclear magnetic resonance frequency of a paramagnetic substance first published in 1949 by the American physicist Walter David Knight.

The Knight shift is due to the conduction electrons in metals. They introduce an "extra" effective field at the nuclear site, due to the spin orientations of the conduction electrons in the presence of an external field. This is responsible for the shift observed in the nuclear magnetic resonance. The shift comes from two sources, one is the Pauli paramagnetic spin susceptibility, the other is the s-component wavefunctions at the nucleus.

Depending on the electronic structure, Knight shift may be temperature dependent. However, in metals which normally have a broad featureless electronic density of states, Knight shifts are temperature independent.

Robinson oscillator

The **Robinson oscillator** (or **Robinson marginal oscillator**) is an electronic circuit used in the field of Nuclear Magnetic Resonance (NMR). The oscillator forms the underlying basis of Magnetic Resonance Imaging (MRI) systems used in many hospitals. It was invented by the British physicist Neville Robinson.

References

- Deschamps, P., Vaissière, J. and Sullivan, N. S., Integrated circuit Robinson oscillator for NMR detection ^[1], *Review of Scientific Instruments*, 48(6):664–668, June 1977. DOI 10.1063/1.1135103
- Wilson, K. J. and Vallabhan, C. P. G., An improved MOSFET-based Robinson oscillator for NMR detection ^[2], *Meas. Sci. Technol.*, 1(5):458-460, May 1990. DOI 10.1088/0957-0233/1/5/015

References

[1] http://content.aip.org/RSINAK/v48/i6/664_1.html

[2] <http://www.iop.org/EJ/abstract/0957-0233/1/5/015>

Fourier transform

In mathematics, the **Fourier transform** (often abbreviated **FT**) is an operation that transforms one complex-valued function of a real variable into another. In such applications as signal processing, the domain of the original function is typically time and is accordingly called the *time domain*. The domain of the new function is typically called the frequency domain, and the new function itself is called the *frequency domain representation* of the original function. It describes which frequencies are present in the original function. This is analogous to describing a musical chord in terms of the notes being played. In effect, the Fourier transform decomposes a function into oscillatory functions. The term Fourier transform refers both to the frequency domain representation of a function, and to the process or formula that "transforms" one function into the other.

The Fourier transform and its generalizations are the subject of Fourier analysis. In this specific case, both the time and frequency domains are unbounded linear continua. It is possible to define the Fourier transform of a function of several variables, which is important for instance in the physical study of wave motion and optics. It is also possible to generalize the Fourier transform on discrete structures such as finite groups, efficient computation of which through a fast Fourier transform is essential for high-speed computing.

Fourier transforms
Continuous Fourier transform
Fourier series
Discrete Fourier transform
Discrete-time Fourier transform
Related transforms

Definition

There are several common conventions for defining the Fourier transform of an integrable function $f : \mathbf{R} \rightarrow \mathbf{C}$ (Kaiser 1994). This article will use the definition:

$$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx, \text{ for every real number } \xi.$$

When the independent variable x represents *time* (with SI unit of seconds), the transform variable ξ represents frequency (in hertz). Under suitable conditions, f can be reconstructed from $\hat{f}$ by the **inverse transform**:

$$f(x) = \int_{-\infty}^{\infty} \hat{f}(\xi) e^{2\pi i x \xi} d\xi, \text{ for every real number } x.$$

For other common conventions and notations, including using the angular frequency ω instead of the frequency ξ , see Other conventions and Other notations below. The Fourier transform on Euclidean space is treated separately, in which the variable x often represents position and ξ momentum.

Introduction

The motivation for the Fourier transform comes from the study of Fourier series. In the study of Fourier series, complicated periodic functions are written as the sum of simple waves mathematically represented by sines and cosines. Due to the properties of sine and cosine it is possible to recover the amount of each wave in the sum by an integral. In many cases it is desirable to use Euler's formula, which states that $e^{2\pi i \theta} = \cos 2\pi \theta + i \sin 2\pi \theta$, to write Fourier series in terms of the basic waves $e^{2\pi i \theta}$. This has the advantage of simplifying many of the formulas involved and providing a formulation for Fourier series that more closely resembles the definition followed in this article. This passage from sines and cosines to complex exponentials makes it necessary for the Fourier coefficients to be complex valued. The usual interpretation of this complex number is that it gives you both the amplitude (or size) of the wave present in the function and the phase (or the initial angle) of the wave. This passage also introduces the need for negative "frequencies". If θ were measured in seconds then the waves $e^{2\pi i \theta}$ and $e^{-2\pi i \theta}$ would both complete one cycle per second, but they represent different frequencies in the Fourier transform. Hence, frequency no longer measures the number of cycles per unit time, but is closely related.

We may use Fourier series to motivate the Fourier transform as follows. Suppose that f is a function which is zero outside of some interval $[-L/2, L/2]$. Then for any $T \geq L$ we may expand f in a Fourier series on the interval $[-T/2, T/2]$, where the "amount" (denoted by c_n) of the wave $e^{2\pi i n x / T}$ in the Fourier series of f is given by

$$\hat{f}(n/T) = c_n = \int_{-T/2}^{T/2} e^{-2\pi i n x / T} f(x) dx$$

and f should be given by the formula

$$f(x) = \frac{1}{T} \sum_{n=-\infty}^{\infty} \hat{f}(n/T) e^{2\pi i n x / T}.$$

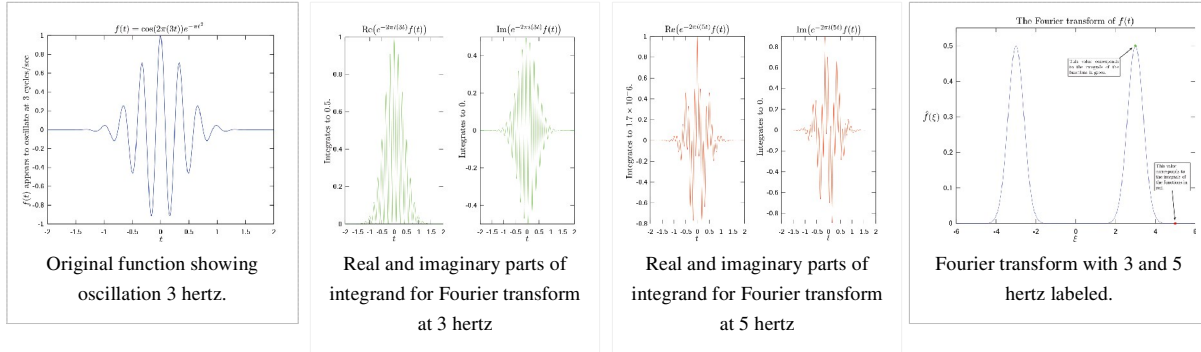
If we let $\xi_n = n/T$, and we let $\Delta\xi = (n+1)/T - n/T = 1/T$, then this last sum becomes the Riemann sum

$$f(x) = \sum_{n=-\infty}^{\infty} \hat{f}(\xi_n) e^{2\pi i x \xi_n} \Delta\xi.$$

By letting $T \rightarrow \infty$ this Riemann sum converges to the integral for the inverse Fourier transform given in the Definition section. Under suitable conditions this argument may be made precise (Stein & Shakarchi 2003). Hence, as in the case of Fourier series, the Fourier transform can be thought of as a function that measures how much of each individual frequency is present in our function, and we can recombine these waves by using an integral (or "continuous sum") to reproduce the original function.

The following images provide a visual illustration of how the Fourier transform measures whether a frequency is present in a particular function. The function depicted $f(t) = \cos(6\pi t) e^{-\pi t^2}$ oscillates at 3 hertz (if t measures

seconds) and tends quickly to 0. This function was specially chosen to have a real Fourier transform which can easily be plotted. The first image contains its graph. In order to calculate $\hat{f}(3)$ we must integrate $e^{-2\pi i(3t)}f(t)$. The second image shows the plot of the real and imaginary parts of this function. The real part of the integrand is almost always positive, this is because when $f(t)$ is negative, then the real part of $e^{-2\pi i(3t)}$ is negative as well. Because they oscillate at the same rate, when $f(t)$ is positive, so is the real part of $e^{-2\pi i(3t)}$. The result is that when you integrate the real part of the integrand you get a relatively large number (in this case 0.5). On the other hand, when you try to measure a frequency that is not present, as in the case when we look at $\hat{f}(5)$, the integrand oscillates enough so that the integral is very small. The general situation may be a bit more complicated than this, but this in spirit is how the Fourier transform measures how much of an individual frequency is present in a function $f(t)$.



Properties of the Fourier transform

An *integrable function* is a function f on the real line that is Lebesgue-measurable and satisfies

$$\int_{-\infty}^{\infty} |f(x)| dx < \infty.$$

Basic properties

Given integrable functions $f(x)$, $g(x)$, and $h(x)$ denote their Fourier transforms by $\hat{f}(\xi)$, $\hat{g}(\xi)$, and $\hat{h}(\xi)$ respectively. The Fourier transform has the following basic properties (Pinsky 2002).

Linearity

For any complex numbers a and b , if $h(x) = af(x) + bg(x)$, then $\hat{h}(\xi) = a \cdot \hat{f}(\xi) + b \cdot \hat{g}(\xi)$.

Translation

For any real number x_0 , if $h(x) = f(x - x_0)$, then $\hat{h}(\xi) = e^{-2\pi i x_0 \xi} \hat{f}(\xi)$.

Modulation

For any real number ξ_0 , if $h(x) = e^{2\pi i x \xi_0} f(x)$, then $\hat{h}(\xi) = \hat{f}(\xi - \xi_0)$.

Scaling

For a non-zero real number a , if $h(x) = f(ax)$, then $\hat{h}(\xi) = \frac{1}{|a|} \hat{f}\left(\frac{\xi}{a}\right)$. The case $a = -1$ leads to the

time-reversal property, which states: if $h(x) = f(-x)$, then $\hat{h}(\xi) = \hat{f}(-\xi)$.

Conjugation

If $h(x) = \overline{f(x)}$, then $\hat{h}(\xi) = \overline{\hat{f}(-\xi)}$.

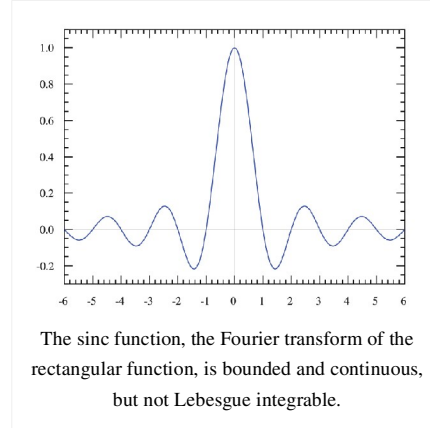
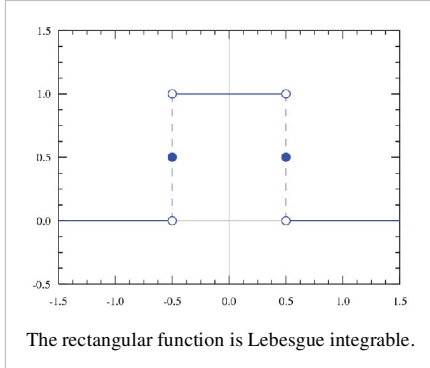
In particular, if f is real, then one has the *reality condition* $\hat{f}(-\xi) = \overline{\hat{f}(\xi)}$.

And if f is purely imaginary, then $\hat{f}(-\xi) = -\overline{\hat{f}(\xi)}$.

Convolution

If $h(x) = (f * g)(x)$, then $\hat{h}(\xi) = \hat{f}(\xi) \cdot \hat{g}(\xi)$.

Uniform continuity and the Riemann–Lebesgue lemma



$$\hat{f}(\xi) \rightarrow 0 \text{ as } |\xi| \rightarrow \infty.$$

The Fourier transform $\hat{f}$ of an integrable function f is bounded and continuous, but need not be integrable – for example, the Fourier transform of the rectangular function, which is a step function (and hence integrable) is the sinc function, which is not Lebesgue integrable, though it does have an improper integral: one has an analog to the alternating harmonic series, which is a convergent sum but not absolutely convergent.

It is not possible in general to write the *inverse transform* as a Lebesgue integral. However, when both f and $\hat{f}$ are integrable, the following inverse equality holds true for almost every x :

$$f(x) = \int_{-\infty}^{\infty} \hat{f}(\xi) e^{2i\pi x \xi} d\xi.$$

Almost everywhere, f is equal to the continuous function given by the right-hand side. If f is given as continuous function on the line, then equality holds for every x .

A consequence of the preceding result is that the Fourier transform is injective on $L^1(\mathbf{R})$.

The Plancherel theorem and Parseval's theorem

Let $f(x)$ and $g(x)$ be integrable, and let $\hat{f}(\xi)$ and $\hat{g}(\xi)$ be their Fourier transforms. If $f(x)$ and $g(x)$ are also square-integrable, then we have Parseval's theorem (Rudin 1987, p. 187):

$$\int_{-\infty}^{\infty} f(x) \overline{g(x)} dx = \int_{-\infty}^{\infty} \hat{f}(\xi) \overline{\hat{g}(\xi)} d\xi,$$

where the bar denotes complex conjugation.

The Plancherel theorem, which is equivalent to Parseval's theorem, states (Rudin 1987, p. 186):

$$\int_{-\infty}^{\infty} |f(x)|^2 dx = \int_{-\infty}^{\infty} |\hat{f}(\xi)|^2 d\xi.$$

The Plancherel theorem makes it possible to define the Fourier transform for functions in $L^2(\mathbf{R})$, as described in Generalizations below. The Plancherel theorem has the interpretation in the sciences that the Fourier transform preserves the energy of the original quantity. It should be noted that depending on the author either of these theorems might be referred to as the Plancherel theorem or as Parseval's theorem.

See Pontryagin duality for a general formulation of this concept in the context of locally compact abelian groups.

Poisson summation formula

The Poisson summation formula provides a link between the study of Fourier transforms and Fourier Series. Given an integrable function f we can consider the periodization of f given by:

$$\bar{f}(x) = \sum_{k \in \mathbb{Z}} f(x + k),$$

where the summation is taken over the set of all integers k . The Poisson summation formula relates the Fourier series of $\bar{f}$ to the Fourier transform of f . Specifically it states that the Fourier series of $\bar{f}$ is given by:

$$\bar{f}(x) \sim \sum_{k \in \mathbb{Z}} \hat{f}(k) e^{2\pi i k x}.$$

Convolution theorem

The Fourier transform translates between convolution and multiplication of functions. If $f(x)$ and $g(x)$ are integrable functions with Fourier transforms $\hat{f}(\xi)$ and $\hat{g}(\xi)$ respectively, then the Fourier transform of the convolution is given by the product of the Fourier transforms $\hat{f}(\xi)$ and $\hat{g}(\xi)$ (under other conventions for the definition of the Fourier transform a constant factor may appear).

This means that if:

$$h(x) = (f * g)(x) = \int_{-\infty}^{\infty} f(y) g(x - y) dy,$$

where $*$ denotes the convolution operation, then:

$$\hat{h}(\xi) = \hat{f}(\xi) \cdot \hat{g}(\xi).$$

In linear time invariant (LTI) system theory, it is common to interpret $g(x)$ as the impulse response of an LTI system with input $f(x)$ and output $h(x)$, since substituting the unit impulse for $f(x)$ yields $h(x) = g(x)$. In this case, $\hat{g}(\xi)$ represents the frequency response of the system.

Conversely, if $f(x)$ can be decomposed as the product of two square integrable functions $p(x)$ and $q(x)$, then the Fourier transform of $f(x)$ is given by the convolution of the respective Fourier transforms $\hat{p}(\xi)$ and $\hat{q}(\xi)$.

Cross-correlation theorem

In an analogous manner, it can be shown that if $h(x)$ is the cross-correlation of $f(x)$ and $g(x)$:

$$h(x) = (f \star g)(x) = \int_{-\infty}^{\infty} \overline{f(y)} g(x + y) dy$$

then the Fourier transform of $h(x)$ is:

$$\hat{h}(\xi) = \overline{\hat{f}(\xi)} \hat{g}(\xi).$$

As a special case, the autocorrelation of function $f(x)$ is:

$$h(x) = (f \star f)(x) = \int_{-\infty}^{\infty} \overline{f(y)} f(x + y) dy$$

for which

$$\hat{h}(\xi) = \overline{\hat{f}(\xi)} \hat{f}(\xi) = |\hat{f}(\xi)|^2.$$

Eigenfunctions

One important choice of an orthonormal basis for $L^2(\mathbf{R})$ is given by the Hermite functions

$$\psi_n(x) = \frac{2^{1/4}}{\sqrt{n!}} e^{-\pi x^2} H_n(2x\sqrt{\pi}),$$

where $H_n(x)$ are the "probabilist's" Hermite polynomials, defined by $H_n(x) = (-1)^n \exp(x^2/2) D^n \exp(-x^2/2)$. Under this convention for the Fourier transform, we have that

$$\hat{\psi}_n(\xi) = (-i)^n \psi_n(\xi).$$

In other words, the Hermite functions form a complete orthonormal system of eigenfunctions for the Fourier transform on $L^2(\mathbf{R})$ (Pinsky 2002). However, this choice of eigenfunctions is not unique. There are only four different eigenvalues of the Fourier transform (± 1 and $\pm i$) and any linear combination of eigenfunctions with the same eigenvalue gives another eigenfunction. As a consequence of this, it is possible to decompose $L^2(\mathbf{R})$ as a direct sum of four spaces H_0 , H_1 , H_2 , and H_3 where the Fourier transform acts on H_k simply by multiplication by i^k . This approach to define the Fourier transform is due to N. Wiener (Duoandikoetxea 2001). The choice of Hermite functions is convenient because they are exponentially localized in both frequency and time domains, and thus give rise to the fractional Fourier transform used in time-frequency analysis (Boashash 2003).

Fourier transform on Euclidean space

The Fourier transform can be in any arbitrary number of dimensions n . As with the one-dimensional case there are many conventions, for an integrable function $f(x)$ this article takes the definition:

$$\hat{f}(\xi) = \mathcal{F}(f)(\xi) = \int_{\mathbb{R}^n} f(x) e^{-2\pi i x \cdot \xi} dx$$

where x and ξ are n -dimensional vectors, and $x \cdot \xi$ is the dot product of the vectors. The dot product is sometimes written as $\langle x, \xi \rangle$.

All of the basic properties listed above hold for the n -dimensional Fourier transform, as do Plancherel's and Parseval's theorem. When the function is integrable, the Fourier transform is still uniformly continuous and the Riemann–Lebesgue lemma holds. (Stein & Weiss 1971)

Uncertainty principle

Generally speaking, the more concentrated $f(x)$ is, the more spread out its Fourier transform $\hat{f}(\xi)$ must be. In particular, the scaling property of the Fourier transform may be seen as saying: if we "squeeze" a function in x , its Fourier transform "stretches out" in ξ . It is not possible to arbitrarily concentrate both a function and its Fourier transform.

The trade-off between the compaction of a function and its Fourier transform can be formalized in the form of an **Uncertainty Principle**, and is formalized by viewing a function and its Fourier transform as conjugate variables with respect to the symplectic form on the time–frequency domain: from the point of view of the linear canonical transformation, the Fourier transform is rotation by 90° in the time–frequency domain, and preserves the symplectic form.

Suppose $f(x)$ is an integrable and square-integrable function. Without loss of generality, assume that $f(x)$ is normalized:

$$\int_{-\infty}^{\infty} |f(x)|^2 dx = 1.$$

It follows from the Plancherel theorem that $\hat{f}(\xi)$ is also normalized.

The spread around $x = 0$ may be measured by the *dispersion about zero* (Pinsky 2002) defined by

$$D_0(f) = \int_{-\infty}^{\infty} x^2 |f(x)|^2 dx.$$

In probability terms, this is the second moment of $|f(x)|^2$ about zero.

The Uncertainty principle states that, if $f(x)$ is absolutely continuous and the functions $x f(x)$ and $f'(x)$ are square integrable, then

$$D_0(f) D_0(\hat{f}) \geq \frac{1}{16\pi^2} \quad (\text{Pinsky 2002}).$$

The equality is attained only in the case $f(x) = C_1 e^{-\pi x^2/\sigma^2}$ (hence $\hat{f}(\xi) = \sigma C_1 e^{-\pi \sigma^2 \xi^2}$) where $\sigma > 0$ is arbitrary and C_1 is such that f is L^2 -normalized (Pinsky 2002). In other words, where f is a (normalized) Gaussian function, centered at zero.

In fact, this inequality implies that:

$$\left(\int_{-\infty}^{\infty} (x - x_0)^2 |f(x)|^2 dx \right) \left(\int_{-\infty}^{\infty} (\xi - \xi_0)^2 |\hat{f}(\xi)|^2 d\xi \right) \geq \frac{1}{16\pi^2}$$

for any x_0, ξ_0 in $\mathbf{R}$ (Stein & Shakarchi 2003).

In quantum mechanics, the momentum and position wave functions are Fourier transform pairs, to within a factor of Planck's constant. With this constant properly taken into account, the inequality above becomes the statement of the Heisenberg uncertainty principle (Stein & Shakarchi 2003).

Spherical harmonics

Let the set of homogeneous harmonic polynomials of degree k on $\mathbf{R}^n$ be denoted by $\mathbf{A}_k$. The set $\mathbf{A}_k$ consists of the solid spherical harmonics of degree k . The solid spherical harmonics play a similar role in higher dimensions to the Hermite polynomials in dimension one. Specifically, if $f(x) = e^{-\pi |x|^2} P(x)$ for some $P(x)$ in $\mathbf{A}_k$, then $\hat{f}(\xi) = i^{-k} f(\xi)$. Let the set $\mathbf{H}_k$ be the closure in $L^2(\mathbf{R}^n)$ of linear combinations of functions of the form $f(|x|)P(x)$ where $P(x)$ is in $\mathbf{A}_k$. The space $L^2(\mathbf{R}^n)$ is then a direct sum of the spaces $\mathbf{H}_k$ and the Fourier transform maps each space $\mathbf{H}_k$ to itself and is possible to characterize the action of the Fourier transform on each space $\mathbf{H}_k$ (Stein & Weiss 1971). Let $f(x) = f_0(|x|)P(x)$ (with $P(x)$ in $\mathbf{A}_k$), then $\hat{f}(\xi) = F_0(|\xi|)P(\xi)$ where

$$F_0(r) = 2\pi i^{-k} r^{-(n+2k-2)/2} \int_0^\infty f_0(s) J_{(n+2k-2)/2}(2\pi r s) s^{(n+2k)/2} ds.$$

Here $J_{(n+2k-2)/2}$ denotes the Bessel function of the first kind with order $(n+2k-2)/2$. When $k=0$ this gives a useful formula for the Fourier transform of a radial function (Grafakos 2004).

Restriction problems

In higher dimensions it becomes interesting to study *restriction problems* for the Fourier transform. The Fourier transform of an integrable function is continuous and the restriction of this function to any set is defined. But for a square-integrable function the Fourier transform could be a general *class* of square integrable functions. As such, the restriction of the Fourier transform of an $L^2(\mathbf{R}^n)$ function cannot be defined on sets of measure 0. It is still an active area of study to understand restriction problems in L^p for $1 < p < 2$. Surprisingly, it is possible in some cases to define the restriction of a Fourier transform to a set S , provided S has non-zero curvature. The case when S is the unit sphere in $\mathbf{R}^n$ is of particular interest. In this case the Tomas-Stein restriction theorem states that the restriction of the Fourier transform to the unit sphere in $\mathbf{R}^n$ is a bounded operator on L^p provided $1 \leq p \leq (2n+2)/(n+3)$.

One notable difference between the Fourier transform in 1 dimension versus higher dimensions concerns the partial sum operator. Consider an increasing collection of measurable sets E_R indexed by $R \in (0, \infty)$: such as balls of radius R centered at the origin, or cubes of side $2R$. For a given integrable function f , consider the function f_R defined by:

$$f_R(x) = \int_{E_R} \hat{f}(\xi) e^{2\pi i x \cdot \xi} d\xi, \quad x \in \mathbf{R}^n.$$

Suppose in addition that f is in $L^p(\mathbf{R}^n)$. For $n = 1$ and $1 < p < \infty$, if one takes $E_R = (-R, R)$, then f_R converges to f in L^p as R tends to infinity, by the boundedness of the Hilbert transform. Naively one may hope the same holds true for $n > 1$. In the case that E_R is taken to be a cube with side length R , then convergence still holds. Another natural candidate is the Euclidean ball $E_R = \{\xi : |\xi| < R\}$. In order for this partial sum operator to converge, it is necessary that the multiplier for the unit ball be bounded in $L^p(\mathbf{R}^n)$. For $n \geq 2$ it is a celebrated theorem of Charles Fefferman that the multiplier for the unit ball is never bounded unless $p = 2$ (Duoandikoetxea 2001). In fact, when $p \neq 2$, this shows that not only may f_R fail to converge to f in L^p , but for some functions $f \in L^p(\mathbf{R}^n)$, f_R is not even an element of L^p .

Generalizations

Fourier transform on other function spaces

It is possible to extend the definition of the Fourier transform to other spaces of functions. Since compactly supported smooth functions are integrable and dense in $L^2(\mathbf{R})$, the Plancherel theorem allows us to extend the definition of the Fourier transform to general functions in $L^2(\mathbf{R})$ by continuity arguments. Further $\mathcal{F} : L^2(\mathbf{R}) \rightarrow L^2(\mathbf{R})$ is a unitary operator (Stein & Weiss 1971, Thm. 2.3). Many of the properties remain the same for the Fourier transform. The Hausdorff–Young inequality can be used to extend the definition of the Fourier transform to include functions in $L^p(\mathbf{R})$ for $1 \leq p \leq 2$. Unfortunately, further extensions become more technical. The Fourier transform of functions in L^p for the range $2 < p < \infty$ requires the study of distributions (Katznelson 1976). In fact, it can be shown that there are functions in L^p with $p > 2$ so that the Fourier transform is not defined as a function (Stein & Weiss 1971).

Fourier–Stieltjes transform

The Fourier transform of a finite Borel measure μ on $\mathbf{R}^n$ is given by (Pinsky 2002):

$$\hat{\mu}(\xi) = \int_{\mathbf{R}^n} e^{-2\pi i x \cdot \xi} d\mu.$$

This transform continues to enjoy many of the properties of the Fourier transform of integrable functions. One notable difference is that the Riemann–Lebesgue lemma fails for measures (Katznelson 1976). In the case that $d\mu = f(x) dx$, then the formula above reduces to the usual definition for the Fourier transform of f . In the case that μ is the probability distribution associated to a random variable X , the Fourier–Stieltjes transform is closely related to the characteristic function, but the typical conventions in probability theory take $e^{ix\xi}$ instead of $e^{-2\pi i x \cdot \xi}$ (Pinsky 2002). In the case when the distribution has a probability density function this definition reduces to the Fourier transform applied to the probability density function, again with a different choice of constants.

The Fourier transform may be used to give a characterization of continuous measures. Bochner's theorem characterizes which functions may arise as the Fourier–Stieltjes transform of a measure (Katznelson 1976).

Furthermore, the Dirac delta function is not a function but it is a finite Borel measure. Its Fourier transform is a constant function (whose specific value depends upon the form of the Fourier transform used).

Tempered distributions

The Fourier transform maps the space of Schwartz functions to itself, and gives a homeomorphism of the space to itself (Stein & Weiss 1971). Because of this it is possible to define the Fourier transform of tempered distributions. These include all the integrable functions mentioned above and have the added advantage that the Fourier transform of any tempered distribution is again a tempered distribution.

The following two facts provide some motivation for the definition of the Fourier transform of a distribution. First let f and g be integrable functions, and let $\hat{f}$ and $\hat{g}$ be their Fourier transforms respectively. Then the Fourier transform obeys the following multiplication formula (Stein & Weiss 1971),

$$\int_{\mathbb{R}^n} \hat{f}(x)g(x) dx = \int_{\mathbb{R}^n} f(x)\hat{g}(x) dx.$$

Secondly, every integrable function f defines a distribution T_f by the relation

$$T_f(\varphi) = \int_{\mathbb{R}^n} f(x)\varphi(x) dx \quad \text{for all Schwartz functions } \varphi.$$

In fact, given a distribution T , we define the Fourier transform by the relation

$$\hat{T}(\varphi) = T(\hat{\varphi}) \quad \text{for all Schwartz functions } \varphi.$$

It follows that

$$\hat{T}_f = T_{\hat{f}}.$$

Distributions can be differentiated and the above mentioned compatibility of the Fourier transform with differentiation and convolution remains true for tempered distributions.

Locally compact abelian groups

The Fourier transform may be generalized to any locally compact abelian group. A locally compact abelian group is an abelian group which is at the same time a locally compact Hausdorff topological space so that the group operations are continuous. If G is a locally compact abelian group, it has a translation invariant measure μ , called Haar measure. For a locally compact abelian group G it is possible to place a topology on the set of characters $\hat{G}$ so that $\hat{G}$ is also a locally compact abelian group. For a function f in $L^1(G)$ it is possible to define the Fourier transform by (Katznelson 1976):

$$\hat{f}(\xi) = \int_G \xi(x)f(x) d\mu \quad \text{for any } \xi \in \hat{G}.$$

Locally compact Hausdorff space

The Fourier transform may be generalized to any locally compact Hausdorff space, which recovers the topology but loses the group structure.

Given a locally compact Hausdorff topological space X , the space $A=C_0(X)$ of continuous complex-valued functions on X which vanish at infinity is in a natural way a commutative C^* -algebra, via pointwise addition, multiplication, complex conjugation, and with norm as the uniform norm. Conversely, the characters of this algebra A , denoted Φ_A , are naturally a topological space, and can be identified with evaluation at a point of x , and one has an isometric isomorphism $C_0(X) \rightarrow C_0(\Phi_A)$. In the case where $X=\mathbf{R}$ is the real line, this is exactly the Fourier transform.

Non-abelian groups

The Fourier transform can also be defined for functions on a non-abelian group, provided that the group is compact. Unlike the Fourier transform on an abelian group, which is scalar-valued, the Fourier transform on a non-abelian group is operator-valued (Hewitt & Ross 1971, Chapter 8). The Fourier transform on compact groups is a major tool in representation theory (Knapp 2001) and non-commutative harmonic analysis.

Let G be a compact Hausdorff topological group. Let Σ denote the collection of all isomorphism classes of finite-dimensional irreducible unitary representations, along with a definite choice of representation $U^{(\sigma)}$ on the Hilbert space H_σ of finite dimension d_σ for each $\sigma \in \Sigma$. If μ is a finite Borel measure on G , then the Fourier–Stieltjes transform of μ is the operator on H_σ defined by

$$\langle \hat{\mu}\xi, \eta \rangle_{H_\sigma} = \int_G \langle \overline{U}_g^{(\sigma)} \xi, \eta \rangle d\mu(g)$$

where $\overline{U}^{(\sigma)}$ is the complex-conjugate representation of $U^{(\sigma)}$ acting on H_σ . As in the abelian case, if μ is absolutely continuous with respect to the left-invariant probability measure λ on G , then it is represented as

$$d\mu = f d\lambda$$

for some $f \in L^1(\lambda)$. In this case, one identifies the Fourier transform of f with the Fourier–Stieltjes transform of μ .

The mapping $\mu \mapsto \hat{\mu}$ defines an isomorphism between the Banach space $M(G)$ of finite Borel measures (see rca space) and a closed subspace of the Banach space $\mathbf{C}_\infty(\Sigma)$ consisting of all sequences $E = (E_\sigma)$ indexed by Σ of (bounded) linear operators $E_\sigma : H_\sigma \rightarrow H_\sigma$ for which the norm

$$\|E\| = \sup_{\sigma \in \Sigma} \|E_\sigma\|$$

is finite. The "convolution theorem" asserts that, furthermore, this isomorphism of Banach spaces is in fact an isomorphism of C^* algebras into a subspace of $\mathbf{C}_\infty(\Sigma)$, in which $M(G)$ is equipped with the product given by convolution of measures and $\mathbf{C}_\infty(\Sigma)$ the product given by multiplication of operators in each index σ .

The Peter-Weyl theorem holds, and a version of the Fourier inversion formula (Plancherel's theorem) follows: if $f \in L^2(G)$, then

$$f(g) = \sum_{\sigma \in \Sigma} d_\sigma \operatorname{tr}(\hat{f}(\sigma) U_g^{(\sigma)})$$

where the summation is understood as convergent in the L^2 sense.

The generalization of the Fourier transform to the noncommutative situation has also in part contributed to the development of noncommutative geometry. In this context, a categorical generalization of the Fourier transform to noncommutative groups is Tannaka-Krein duality, which replaces the group of characters with the category of representations. However, this loses the connection with harmonic functions.

Alternatives

In signal processing terms, a function (of time) is a representation of a signal with perfect *time resolution*, but no frequency information, while the Fourier transform has perfect *frequency resolution*, but no time information: the magnitude of the Fourier transform at a point is how much frequency content there is, but location is only given by phase (argument of the Fourier transform at a point), and standing waves are not localized in time – a sine wave continues out to infinity, without decaying. This limits the usefulness of the Fourier transform for analyzing signals that are localized in time, notably transients, or any signal of finite extent.

As alternatives to the Fourier transform, in time-frequency analysis, one uses time-frequency transforms or time-frequency distributions to represent signals in a form that has some time information and some frequency information – by the uncertainty principle, there is a trade-off between these. These can be generalizations of the Fourier transform, such as the short-time Fourier transform or fractional Fourier transform, or can use different functions to represent signals, as in wavelet transforms and chirplet transforms, with the wavelet analog of the

(continuous) Fourier transform being the continuous wavelet transform. (Boashash 2003).

Applications

Analysis of differential equations

Fourier transforms and the closely related Laplace transforms are widely used in solving differential equations. The Fourier transform is compatible with differentiation in the following sense: if $f(x)$ is a differentiable function with Fourier transform $\hat{f}(\xi)$, then the Fourier transform of its derivative is given by $2\pi i \xi \hat{f}(\xi)$. This can be used to transform differential equations into algebraic equations. Note that this technique only applies to problems whose domain is the whole set of real numbers. By extending the Fourier transform to functions of several variables partial differential equations with domain $\mathbf{R}^n$ can also be translated into algebraic equations.

FT-NMR, FT-IR, FT-NIR and MRI

The Fourier transform is also used in nuclear magnetic resonance (NMR) and in other kinds of spectroscopy, e.g. infrared (FT-IR). In NMR an exponentially-shaped free induction decay (FID) signal is acquired in the time domain and Fourier-transformed to a Lorentzian line-shape in the frequency domain. The Fourier transform is also used in magnetic resonance imaging (MRI) and mass spectrometry.

Domain and range of the Fourier transform

It is often desirable to have the most general domain for the Fourier transform as possible. The definition of Fourier transform as an integral naturally restricts the domain to the space of integrable functions. Unfortunately, there is no simple characterizations of which functions are Fourier transforms of integrable functions (Stein & Weiss 1971). It is possible to extend the domain of the Fourier transform in various ways, as discussed in generalizations above. The following list details some of the more common domains and ranges on which the Fourier transform is defined.

- The space of Schwartz functions is closed under the Fourier transform. Schwartz functions are rapidly decaying functions and do not include all functions which are relevant for the Fourier transform. More details may be found in (Stein & Weiss 1971).
- The space L^p maps into the space L^q , where $1/p + 1/q = 1$ and $1 \leq p \leq 2$ (Hausdorff–Young inequality).
- In particular, the space L^2 is closed under the Fourier transform, but here the Fourier transform is no longer defined by integration.
- The space L^1 of Lebesgue integrable functions maps into C_0 , the space of continuous functions that tend to zero at infinity – not just into the space L^∞ of bounded functions (the Riemann–Lebesgue lemma).
- The set of tempered distributions is closed under the Fourier transform. Tempered distributions are also a form of generalization of functions. It is in this generality that one can define the Fourier transform of objects like the Dirac comb.

Other notations

Other common notations for $\hat{f}(\xi)$ are: $\tilde{f}(\xi)$, $F(\xi)$, $\mathcal{F}(f)(\xi)$, $(\mathcal{F}f)(\xi)$, $\mathcal{F}(f)$, $\mathcal{F}(\omega)$, $\mathcal{F}(j\omega)$, $\mathcal{F}\{f\}$ and $\mathcal{F}(f(t))$. Though less commonly other notations are used. Denoting the Fourier transform by a capital letter corresponding to the letter of function being transformed (such as $f(x)$ and $F(\xi)$) is especially common in the sciences and engineering. In electronics, the omega (ω) is often used instead of ξ due to its interpretation as angular frequency, sometimes it is written as $F(j\omega)$, where j is the imaginary unit, to indicate its relationship with the Laplace transform, and sometimes it is written informally as $F(2\pi f)$ in order to use ordinary frequency.

The interpretation of the complex function $\hat{f}(\xi)$ may be aided by expressing it in polar coordinate form:

$\hat{f}(\xi) = A(\xi)e^{i\varphi(\xi)}$ in terms of the two real functions $A(\xi)$ and $\varphi(\xi)$ where:

$$A(\xi) = |\hat{f}(\xi)|,$$

is the amplitude and

$$\varphi(\xi) = \arg(\hat{f}(\xi)),$$

is the phase (see $\arg$ function).

Then the inverse transform can be written:

$$f(x) = \int_{-\infty}^{\infty} A(\xi) e^{i(2\pi\xi x + \varphi(\xi))} d\xi,$$

which is a recombination of all the **frequency components** of $f(x)$. Each component is a complex sinusoid of the form $e^{2\pi i x \xi}$ whose amplitude is $A(\xi)$ and whose initial phase angle (at $x = 0$) is $\varphi(\xi)$.

The Fourier transform may be thought of as a mapping on function spaces. This mapping is here denoted $\mathcal{F}$ and $\mathcal{F}(f)$ is used to denote the Fourier transform of the function f . This mapping is linear, which means that $\mathcal{F}$ can also be seen as a linear transformation on the function space and implies that the standard notation in linear algebra of applying a linear transformation to a vector (here the function f) can be used to write $\mathcal{F}f$ instead of $\mathcal{F}(f)$.

Since the result of applying the Fourier transform is again a function, we can be interested in the value of this function evaluated at the value ξ for its variable, and this is denoted either as $\mathcal{F}(f)(\xi)$ or as $(\mathcal{F}f)(\xi)$. Notice that in the former case, it is implicitly understood that $\mathcal{F}$ is applied first to f and then the resulting function is evaluated at ξ , not the other way around.

In mathematics and various applied sciences it is often necessary to distinguish between a function f and the value of f when its variable equals x , denoted $f(x)$. This means that a notation like $\mathcal{F}(f(x))$ formally can be interpreted as the Fourier transform of the values of f at x . Despite this flaw, the previous notation appears frequently, often when a particular function or a function of a particular variable is to be transformed. For example, $\mathcal{F}(\text{rect}(x)) = \text{sinc}(\xi)$ is sometimes used to express that the Fourier transform of a rectangular function is a sinc function, or $\mathcal{F}(f(x + x_0)) = \mathcal{F}(f(x))e^{2\pi i \xi x_0}$ is used to express the shift property of the Fourier transform. Notice, that the last example is only correct under the assumption that the transformed function is a function of x , not of x_0 .

Other conventions

There are three common conventions for defining the Fourier transform. The Fourier transform is often written in terms of angular frequency: $\omega = 2\pi\xi$ whose units are radians per second.

The substitution $\xi = \omega/(2\pi)$ into the formulas above produces this convention:

$$\hat{f}(\omega) = \int_{\mathbb{R}^n} f(x) e^{-i\omega \cdot x} dx.$$

Under this convention, the inverse transform becomes:

$$f(x) = \frac{1}{(2\pi)^n} \int_{\mathbb{R}^n} \hat{f}(\omega) e^{i\omega \cdot x} d\omega.$$

Unlike the convention followed in this article, when the Fourier transform is defined this way, it is no longer a unitary transformation on $L^2(\mathbf{R}^n)$. There is also less symmetry between the formulas for the Fourier transform and its inverse.

Another popular convention is to split the factor of $(2\pi)^n$ evenly between the Fourier transform and its inverse, which leads to definitions:

$$\hat{f}(\omega) = \frac{1}{(2\pi)^{n/2}} \int_{\mathbb{R}^n} f(x) e^{-i\omega \cdot x} dx$$

$$f(x) = \frac{1}{(2\pi)^{n/2}} \int_{\mathbb{R}^n} \hat{f}(\omega) e^{i\omega \cdot x} d\omega.$$

Under this convention, the Fourier transform is again a unitary transformation on $L^2(\mathbb{R}^n)$. It also restores the symmetry between the Fourier transform and its inverse.

Variations of all three conventions can be created by conjugating the complex-exponential kernel of both the forward and the reverse transform. The signs must be opposites. Other than that, the choice is (again) a matter of convention.

Summary of popular forms of the Fourier transform

ordinary frequency ξ (hertz)	unitary	$\hat{f}_1(\xi) \stackrel{\text{def}}{=} \int_{\mathbb{R}^n} f(x) e^{-2\pi i x \cdot \xi} dx = \hat{f}_2(2\pi\xi) = (2\pi)^{n/2} \hat{f}_3(2\pi\xi)$ $f(x) = \int_{\mathbb{R}^n} \hat{f}_1(\xi) e^{2\pi i x \cdot \xi} d\xi$
angular frequency ω (rad/s)	non-unitary	$\hat{f}_2(\omega) \stackrel{\text{def}}{=} \int_{\mathbb{R}^n} f(x) e^{-i\omega \cdot x} dx = \hat{f}_1\left(\frac{\omega}{2\pi}\right) = (2\pi)^{n/2} \hat{f}_3(\omega)$ $f(x) = \frac{1}{(2\pi)^n} \int_{\mathbb{R}^n} \hat{f}_2(\omega) e^{i\omega \cdot x} d\omega$
	unitary	$\hat{f}_3(\omega) \stackrel{\text{def}}{=} \frac{1}{(2\pi)^{n/2}} \int_{\mathbb{R}^n} f(x) e^{-i\omega \cdot x} dx = \frac{1}{(2\pi)^{n/2}} \hat{f}_1\left(\frac{\omega}{2\pi}\right) = \frac{1}{(2\pi)^{n/2}} \hat{f}_2(\omega)$ $f(x) = \frac{1}{(2\pi)^{n/2}} \int_{\mathbb{R}^n} \hat{f}_3(\omega) e^{i\omega \cdot x} d\omega$

The ordinary-frequency convention (which is used in this article) is the one most often found in the mathematics literature. In the physics literature, the two angular-frequency conventions are more commonly used.

As discussed above, the characteristic function of a random variable is the same as the Fourier–Stieltjes transform of its distribution measure, but in this context it is typical to take a different convention for the constants. Typically characteristic function is defined $E(e^{it \cdot X}) = \int e^{it \cdot x} d\mu_X(x)$. As in the case of the "non-unitary angular frequency" convention above, there is no factor of 2π appearing in either of the integral, or in the exponential. Unlike any of the conventions appearing above, this convention takes the opposite sign in the exponential.

Tables of important Fourier transforms

The following tables record some closed form Fourier transforms. For functions $f(x)$, $g(x)$ and $h(x)$ denote their Fourier transforms by $\hat{f}$, $\hat{g}$, and $\hat{h}$ respectively. Only the three most common conventions are included. It is sometimes useful to notice that entry 105 gives a relationship between the Fourier transform of a function and the original function, which can be seen as relating the Fourier transform and its inverse.

Functional relationships

The Fourier transforms in this table may be found in (Erdélyi 1954) or the appendix of (Kammler 2000).

	Function	Fourier transform unitary, ordinary frequency	Fourier transform unitary, angular frequency	Fourier transform non-unitary, angular frequency	Remarks
	$f(x)$	$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx$	$\hat{f}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx$	$\hat{f}(\nu) = \int_{-\infty}^{\infty} f(x) e^{-i\nu x} dx$	Definition
101	$a \cdot f(x) + b \cdot g(x)$	$a \cdot \hat{f}(\xi) + b \cdot \hat{g}(\xi)$	$a \cdot \hat{f}(\omega) + b \cdot \hat{g}(\omega)$	$a \cdot \hat{f}(\nu) + b \cdot \hat{g}(\nu)$	Linearity
102	$f(x - a)$	$e^{-2\pi i a \xi} \hat{f}(\xi)$	$e^{-i a \omega} \hat{f}(\omega)$	$e^{-i a \nu} \hat{f}(\nu)$	Shift in time domain
103	$e^{2\pi i a x} f(x)$	$\hat{f}(\xi - a)$	$\hat{f}(\omega - 2\pi a)$	$\hat{f}(\nu - 2\pi a)$	Shift in frequency domain, dual of 102
104	$f(ax)$	$\frac{1}{ a } \hat{f}\left(\frac{\xi}{a}\right)$	$\frac{1}{ a } \hat{f}\left(\frac{\omega}{a}\right)$	$\frac{1}{ a } \hat{f}\left(\frac{\nu}{a}\right)$	Scaling in the time domain. If $ a $ is large, then $f(ax)$ is concentrated around 0 and $\frac{1}{ a } \hat{f}\left(\frac{\omega}{a}\right)$ spreads out and flattens.
105	$\hat{f}(x)$	$f(-\xi)$	$f(-\omega)$	$2\pi f(-\nu)$	Duality. Here $\hat{f}$ needs to be calculated using the same method as Fourier transform column. Results from swapping "dummy" variables of x and ξ or ω or ν .
106	$\frac{d^n f(x)}{dx^n}$	$(2\pi i \xi)^n \hat{f}(\xi)$	$(i\omega)^n \hat{f}(\omega)$	$(i\nu)^n \hat{f}(\nu)$	
107	$x^n f(x)$	$\left(\frac{i}{2\pi}\right)^n \frac{d^n \hat{f}(\xi)}{d\xi^n}$	$i^n \frac{d^n \hat{f}(\omega)}{d\omega^n}$	$i^n \frac{d^n \hat{f}(\nu)}{d\nu^n}$	This is the dual of 106

108	$(f * g)(x)$	$\hat{f}(\xi)\hat{g}(\xi)$	$\sqrt{2\pi}\hat{f}(\omega)\hat{g}(\omega)$	$\hat{f}(\nu)\hat{g}(\nu)$	The notation $f * g$ denotes the convolution of f and g — this rule is the convolution theorem
109	$f(x)g(x)$	$(\hat{f} * \hat{g})(\xi)$	$\frac{(\hat{f} * \hat{g})(\omega)}{\sqrt{2\pi}}$	$\frac{1}{2\pi}(\hat{f} * \hat{g})(\nu)$	This is the dual of 108
110	For $f(x)$ a purely real	$\hat{f}(-\xi) = \overline{\hat{f}(\xi)}$	$\hat{f}(-\omega) = \overline{\hat{f}(\omega)}$	$\hat{f}(-\nu) = \overline{\hat{f}(\nu)}$	Hermitian symmetry. $\overline{}$ indicates the complex conjugate.
111	For $f(x)$ a purely real even function	$\hat{f}(\omega)$, $\hat{f}(\xi)$ and $\hat{f}(\nu)$ are purely real even functions.			
112	For $f(x)$ a purely real odd function	$\hat{f}(\omega)$, $\hat{f}(\xi)$ and $\hat{f}(\nu)$ are purely imaginary odd functions.			

Square-integrable functions

The Fourier transforms in this table may be found in (Campbell & Foster 1948), (Erdélyi 1954), or the appendix of (Kammler 2000).

	Function	Fourier transform unitary, ordinary frequency	Fourier transform unitary, angular frequency	Fourier transform non-unitary, angular frequency	Remarks
	$f(x)$	$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i x \xi} dx$	$\hat{f}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x) e^{-i\omega x} dx$	$\hat{f}(\nu) = \int_{-\infty}^{\infty} f(x) e^{-i\nu x} dx$	
201	$\text{rect}(ax)$	$\frac{1}{ a } \cdot \text{sinc}\left(\frac{\xi}{a}\right)$	$\frac{1}{\sqrt{2\pi}a^2} \cdot \text{sinc}\left(\frac{\omega}{2\pi a}\right)$	$\frac{1}{ a } \cdot \text{sinc}\left(\frac{\nu}{2\pi a}\right)$	The rectangular pulse and the <i>normalized</i> sinc function, here defined as $\text{sinc}(x) = \sin(\pi x)/(\pi x)$
202	$\text{sinc}(ax)$	$\frac{1}{ a } \cdot \text{rect}\left(\frac{\xi}{a}\right)$	$\frac{1}{\sqrt{2\pi}a^2} \cdot \text{rect}\left(\frac{\omega}{2\pi a}\right)$	$\frac{1}{ a } \cdot \text{rect}\left(\frac{\nu}{2\pi a}\right)$	Dual of rule 201. The rectangular function is an ideal low-pass filter, and the sinc function is the non-causal impulse response of such a filter.

203	$\text{sinc}^2(ax)$	$\frac{1}{ a } \cdot \text{tri}\left(\frac{\xi}{a}\right)$	$\frac{1}{\sqrt{2\pi}a^2} \cdot \text{tri}\left(\frac{\omega}{2\pi a}\right)$	$\frac{1}{ a } \cdot \text{tri}\left(\frac{\nu}{2\pi a}\right)$	The function $\text{tri}(x)$ is the triangular function
204	$\text{tri}(ax)$	$\frac{1}{ a } \cdot \text{sinc}^2\left(\frac{\xi}{a}\right)$	$\frac{1}{\sqrt{2\pi}a^2} \cdot \text{sinc}^2\left(\frac{\omega}{2\pi a}\right)$	$\frac{1}{ a } \cdot \text{sinc}^2\left(\frac{\nu}{2\pi a}\right)$	Dual of rule 203.
205	$e^{-ax}u(x)$	$\frac{1}{a + 2\pi i\xi}$	$\frac{1}{\sqrt{2\pi}(a + i\omega)}$	$\frac{1}{a + i\nu}$	The function $u(x)$ is the Heaviside unit step function and $a > 0$.
206	$e^{-\alpha x^2}$	$\sqrt{\frac{\pi}{\alpha}} \cdot e^{-\frac{(\pi\xi)^2}{\alpha}}$	$\frac{1}{\sqrt{2\alpha}} \cdot e^{-\frac{\omega^2}{4\alpha}}$	$\sqrt{\frac{\pi}{\alpha}} \cdot e^{-\frac{\nu^2}{4\alpha}}$	This shows that, for the unitary Fourier transforms, the Gaussian function $\exp(-\alpha x^2)$ is its own Fourier transform for some choice of α . For this to be integrable we must have $\text{Re}(\alpha) > 0$.
207	$e^{-a x }$	$\frac{2a}{a^2 + 4\pi^2\xi^2}$	$\sqrt{\frac{2}{\pi}} \cdot \frac{a}{a^2 + \omega^2}$	$\frac{2a}{a^2 + \nu^2}$	For $a > 0$. That is, the Fourier transform of a decaying exponential function is a Lorentzian function.
208	$\frac{J_n(x)}{x}$	$\frac{2i}{n}(-i)^n \cdot U_{n-1}(2\pi\xi) \cdot \sqrt{1 - 4\pi^2\xi^2} \text{rect}(\pi\xi)$	$\sqrt{\frac{2}{\pi}} \frac{i}{n}(-i)^n \cdot U_{n-1}(\omega) \cdot \sqrt{1 - \omega^2} \text{rect}\left(\frac{\omega}{2}\right)$	$\frac{2i}{n}(-i)^n \cdot U_{n-1}(\nu) \cdot \sqrt{1 - \nu^2} \text{rect}\left(\frac{\nu}{2}\right)$	The functions $J_n(x)$ are the n -th order Bessel functions of the first kind. The functions $U_n(x)$ are the Chebyshev polynomial of the second kind. See 315 and 316 below.
209	$\text{sech}(ax)$	$\frac{\pi}{a} \text{sech}\left(\frac{\pi^2}{a}\xi\right)$	$\frac{1}{a} \sqrt{\frac{\pi}{2}} \text{sech}\left(\frac{\pi}{2a}\omega\right)$	$\frac{\pi}{a} \text{sech}\left(\frac{\pi}{2a}\nu\right)$	Hyperbolic secant is its own Fourier transform

Distributions

The Fourier transforms in this table may be found in (Erdélyi 1954) or the appendix of (Kammler 2000).

Function	Fourier transform unitary, ordinary frequency	Fourier transform unitary, angular frequency	Fourier transform non-unitary, angular frequency	Remarks
$f(x)$	$\hat{f}(\xi) = \int_{-\infty}^{\infty} f(x)e^{-2\pi i x \xi} dx$	$\hat{f}(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} f(x)e^{-i\omega x} dx$	$\hat{f}(\nu) = \int_{-\infty}^{\infty} f(x)e^{-i\nu x} dx$	
$\delta(x)$	$\delta(\xi)$	$\sqrt{2\pi} \cdot \delta(\omega)$	$2\pi\delta(\nu)$	The distribution $\delta(\xi)$ denotes the Dirac delta function.
1	1	$\frac{1}{\sqrt{2\pi}}$	1	Dual of rule 301.
e^{iax}	$\delta\left(\xi - \frac{a}{2\pi}\right)$	$\sqrt{2\pi} \cdot \delta(\omega - a)$	$2\pi\delta(\nu - a)$	This follows from 103 and 301.
$\cos(ax)$	$\frac{\delta\left(\xi - \frac{a}{2\pi}\right) + \delta\left(\xi + \frac{a}{2\pi}\right)}{2}$	$\sqrt{2\pi} \cdot \frac{\delta(\omega - a) + \delta(\omega + a)}{2}$	$\pi(\delta(\nu - a) + \delta(\nu + a))$	This follows from rules 101 and 303 using Euler's formula: $\cos(ax) = (e^{iax} + e^{-iax})/2$
$\sin(ax)$	$i \cdot \frac{\delta\left(\xi + \frac{a}{2\pi}\right) - \delta\left(\xi - \frac{a}{2\pi}\right)}{2}$	$i\sqrt{2\pi} \cdot \frac{\delta(\omega + a) - \delta(\omega - a)}{2}$	$i\pi(\delta(\nu + a) - \delta(\nu - a))$	This follows from 101 and 303 using $\sin(ax) = (e^{iax} - e^{-iax})/(2i)$.
$\cos(ax^2)$	$\sqrt{\frac{\pi}{a}} \cos\left(\frac{\pi^2 \xi^2}{a} - \frac{\pi}{4}\right)$	$\frac{1}{\sqrt{2a}} \cos\left(\frac{\omega^2}{4a} - \frac{\pi}{4}\right)$	$\sqrt{\frac{\pi}{a}} \cos\left(\frac{\nu^2}{4a} - \frac{\pi}{4}\right)$	
$\sin(ax^2)$	$-\sqrt{\frac{\pi}{a}} \sin\left(\frac{\pi^2 \xi^2}{a} - \frac{\pi}{4}\right)$	$\frac{-1}{\sqrt{2a}} \sin\left(\frac{\omega^2}{4a} - \frac{\pi}{4}\right)$	$-\sqrt{\frac{\pi}{a}} \sin\left(\frac{\nu^2}{4a} - \frac{\pi}{4}\right)$	
x^n	$\left(\frac{i}{2\pi}\right)^n \delta^{(n)}(\xi)$	$i^n \sqrt{2\pi} \delta^{(n)}(\omega)$	$2\pi i^n \delta^{(n)}(\nu)$	Here, n is a natural number and $\delta^{(n)}(\xi)$ is the distribution derivative of the Dirac delta function. This rule follows from rules 107 and 301. Combining this rule with 101, we can transform all polynomials.
$\frac{1}{x}$	$-i\pi \operatorname{sgn}(\xi)$	$-i\sqrt{\frac{\pi}{2}} \operatorname{sgn}(\omega)$	$-i\pi \operatorname{sgn}(\nu)$	Here $\operatorname{sgn}(\xi)$ is the sign function. Note that $1/x$ is a distribution. It is necessary to use the Cauchy principal value when testing against Schwartz functions. This rule is useful in studying the Hilbert transform.
$\frac{1}{x^n} := \frac{(-1)^{n-1}}{(n-1)!} \frac{d^{n-1}}{dx^{n-1}} \log x $	$-i\pi \frac{(-2\pi i \xi)^{n-1}}{(n-1)!} \operatorname{sgn}(\xi)$	$-i\sqrt{\frac{\pi}{2}} \cdot \frac{(-i\omega)^{n-1}}{(n-1)!} \operatorname{sgn}(\omega)$	$-i\pi \frac{(-i\nu)^{n-1}}{(n-1)!} \operatorname{sgn}(\nu)$	$1/x^n$ is the homogeneous distribution defined by regularizing the singularity via $\frac{1}{x^n}[\phi] = \int_{-\infty}^{\infty} \frac{\phi(x) - \sum_{k=0}^{n-1} x^k \phi^{(k)}(0)/k!}{x^n} dx$
$ x ^\alpha$	$-2 \frac{\sin(\pi\alpha/2)\Gamma(\alpha+1)}{ 2\pi\xi ^{\alpha+1}}$	$\frac{-2 \sin(\pi\alpha/2)\Gamma(\alpha+1)}{\sqrt{2\pi} \omega ^{\alpha+1}}$	$-2 \frac{\sin(\pi\alpha/2)\Gamma(\alpha+1)}{ \omega ^{\alpha+1}}$	If $\operatorname{Re} \alpha > -1$, then $ x ^\alpha$ is a locally integrable function, and so a tempered distribution. The function $\alpha \mapsto x ^\alpha$ is a holomorphic function from the right half-plane to the space of tempered distributions. It admits a unique meromorphic extension to a tempered distribution, also denoted $ x ^\alpha$ for $\alpha \neq -2, -4, \dots$ (See homogeneous distribution.)

$n(x)$	$\frac{1}{i\pi\xi}$	$\sqrt{\frac{2}{\pi}} \cdot \frac{1}{i\omega}$	$\frac{2}{i\nu}$	The dual of rule 309. This time the Fourier transforms need to be considered as Cauchy principal value.
$u(x)$	$\frac{1}{2} \left(\frac{1}{i\pi\xi} + \delta(\xi) \right)$	$\sqrt{\frac{\pi}{2}} \left(\frac{1}{i\pi\omega} + \delta(\omega) \right)$	$\pi \left(\frac{1}{i\pi\nu} + \delta(\nu) \right)$	The function $u(x)$ is the Heaviside unit step function. This result follows from rules 101, 301, and 312.
$\sum_{n=-\infty}^{\infty} \delta(x - nT)$	$\frac{1}{T} \sum_{k=-\infty}^{\infty} \delta\left(\xi - \frac{k}{T}\right)$	$\frac{\sqrt{2\pi}}{T} \sum_{k=-\infty}^{\infty} \delta\left(\omega - \frac{2\pi k}{T}\right)$	$\frac{2\pi}{T} \sum_{k=-\infty}^{\infty} \delta\left(\nu - \frac{2\pi k}{T}\right)$	This function is known as the Dirac comb function. This result can be derived from 302 and 102, together with the fact that $\sum_{n=-\infty}^{\infty} e^{inx} = 2\pi \sum_{k=-\infty}^{\infty} \delta(x + 2\pi k)$; see the distributions.
$J_0(x)$	$\frac{2 \operatorname{rect}(\pi\xi)}{\sqrt{1 - 4\pi^2\xi^2}}$	$\sqrt{\frac{2}{\pi}} \cdot \frac{\operatorname{rect}\left(\frac{\omega}{2}\right)}{\sqrt{1 - \omega^2}}$	$\frac{2 \operatorname{rect}\left(\frac{\nu}{2}\right)}{\sqrt{1 - \nu^2}}$	The function $J_0(x)$ is the zeroth order Bessel function of first kind.
$J_n(x)$	$\frac{2(-i)^n T_n(2\pi\xi) \operatorname{rect}(\pi\xi)}{\sqrt{1 - 4\pi^2\xi^2}}$	$\sqrt{\frac{2}{\pi}} \frac{(-i)^n T_n(\omega) \operatorname{rect}\left(\frac{\omega}{2}\right)}{\sqrt{1 - \omega^2}}$	$\frac{2(-i)^n T_n(\nu) \operatorname{rect}\left(\frac{\nu}{2}\right)}{\sqrt{1 - \nu^2}}$	This is a generalization of 315. The function J_n is the n -th order Bessel function of first kind. The function $T_n(x)$ is the Chebyshev polynomial of first kind.

Two-dimensional functions

Function	Fourier transform unitary, ordinary frequency	Fourier transform unitary, angular frequency	Fourier transform non-unitary, angular frequency	Remarks
$f(x, y)$	$\hat{f}(\xi_x, \xi_y) = \iint f(x, y) e^{-2\pi i(\xi_x x + \xi_y y)} dx dy$	$\hat{f}(\omega_x, \omega_y) = \frac{1}{2\pi} \iint f(x, y) e^{-i(\omega_x x + \omega_y y)} dx dy$	$\hat{f}(\nu_x, \nu_y) = \iint f(x, y) e^{-i(\nu_x x + \nu_y y)} dx dy$	The variables $\xi_x, \xi_y, \omega_x, \omega_y$ are real numbers. The integrals are taken over the entire plane.
$e^{-\pi(a^2 x^2 + b^2 y^2)}$	$\frac{1}{ ab } e^{-\pi(\xi_x^2/a^2 + \xi_y^2/b^2)}$	$\frac{1}{2\pi \cdot ab } e^{-\frac{(\omega_x^2/a^2 + \omega_y^2/b^2)}{4\pi}}$	$\frac{1}{ ab } e^{-\frac{(\nu_x^2/a^2 + \nu_y^2/b^2)}{4\pi}}$	Both functions are Gaussian which may have volume.

$\text{circ}(\sqrt{x^2+y^2})$	$\frac{J_1\left(2\pi\sqrt{\xi_x^2+\xi_y^2}\right)}{\sqrt{\xi_x^2+\xi_y^2}}$	$\frac{J_1\left(\sqrt{\omega_x^2+\omega_y^2}\right)}{\sqrt{\omega_x^2+\omega_y^2}}$	$\frac{2\pi J_1\left(\sqrt{\nu_x^2+\nu_y^2}\right)}{\sqrt{\nu_x^2+\nu_y^2}}$	The func is de by circ($0\leq r\leq 1$) and othe This is the distr and expr usin (the 1 Be func of th first kind (Ste: Wei 197: Thn IV.3
-------------------------------	---	---	--	--

Formulas for general *n*-dimensional functions

	Function	Fourier transform unitary, ordinary frequency	Fourier transform unitary, angular frequency	Fourier transform non-unitary, angular frequency	Remarks
	$f(x)$	$\hat{f}(\xi)=\int_{\mathbb{R}^n}f(x)e^{-2\pi i x\cdot \xi}d^nx$	$\hat{f}(\omega)=\frac{1}{(2\pi)^{(n/2)}}\int_{\mathbb{R}^n}f(x)e^{-i\omega\cdot x}d^nx$	$\hat{f}(\nu)=\int_{\mathbb{R}^n}f(x)e^{-ix\cdot \nu}d^nx$	

501	$\chi_{[0,1]}(x)(1 - x ^2)^\delta$	$\pi^{-\delta}\Gamma(\delta + 1) \xi ^{-(n/2)-\delta}$ $\cdot J_{n/2+\delta}(2\pi \xi)$	$2^{-\delta}\Gamma(\delta + 1) \omega ^{-(n/2)-\delta}$ $\cdot J_{n/2+\delta}(\omega)$	$\pi^{-\delta}\Gamma(\delta + 1)\left \frac{\nu}{2\pi}\right ^{-(n/2)-\delta}$ $\cdot J_{n/2+\delta}(\nu)$	The function $\chi_{[0,1]}$ is the indicator function of the interval $[0,1]$. The function $\Gamma(x)$ is the gamma function. The function $J_{n/2+\delta}^a$ is a Bessel function of the first kind with order $n/2+\delta$. Taking $n = 2$ and $\delta = 0$ produces 402. (Stein & Weiss 1971, Thm. 4.13)
502	$ x ^{-\alpha}, \quad 0 < \operatorname{Re} \alpha < n.$	$c_\alpha \xi ^{-(n-\alpha)}$			See Riesz potential. The formula also holds for all $\alpha \neq -n, -n-1, \dots$ by analytic continuation, but then the function and its Fourier transform need to be understood as suitably regularized tempered distributions. See homogeneous distribution.

See also

- Fourier series
- Fast Fourier transform
- Laplace transform
- Discrete Fourier transform
 - DFT matrix
- Discrete-time Fourier transform
- Fourier–Deligne transform
- Fractional Fourier transform
- Linear canonical transform
- Fourier sine transform
- Short-time Fourier transform
- Fourier inversion theorem
- Analog signal processing
- Transform (mathematics)
- Integral transform
 - Hartley transform
 - Hankel transform

References

- Boashash, B., ed. (2003), *Time-Frequency Signal Analysis and Processing: A Comprehensive Reference*, Oxford: Elsevier Science
- Bochner S., Chandrasekharan K. (1949), *Fourier Transforms*, Princeton University Press
- Bracewell, R. N. (2000), *The Fourier Transform and Its Applications* (3rd ed.), Boston: McGraw-Hill.
- Campbell, George; Foster, Ronald (1948), *Fourier Integrals for Practical Applications*, New York: D. Van Nostrand Company, Inc..
- Duoandikoetxea, Javier (2001), *Fourier Analysis*, American Mathematical Society, ISBN 0-8218-2172-5.
- Dym, H; McKean, H (1985), *Fourier Series and Integrals*, Academic Press, ISBN 978-0122264511.
- Erdélyi, Arthur, ed. (1954), *Tables of Integral Transforms*, **1**, New Your: McGraw-Hill
- Fourier, J. B. Joseph (1822), *Théorie Analytique de la Chaleur*^[1], Paris
- Grafakos, Loukas (2004), *Classical and Modern Fourier Analysis*, Prentice-Hall, ISBN 0-13-035399-X.
- Hewitt, Edwin; Ross, Kenneth A. (1970), *Abstract harmonic analysis. Vol. II: Structure and analysis for compact groups. Analysis on locally compact Abelian groups*, Die Grundlehren der mathematischen Wissenschaften, Band 152, Berlin, New York: Springer-Verlag, MR0262773.
- Hörmander, L. (1976), *Linear Partial Differential Operators, Volume 1*, Springer-Verlag, ISBN 978-3540006626.
- James, J.F. (2002), *A Student's Guide to Fourier Transforms* (2nd ed.), New York: Cambridge University Press, ISBN 0-521-00428-4.
- Kaiser, Gerald (1994), *A Friendly Guide to Wavelets*, Birkhäuser, ISBN 0-8176-3711-7
- Kammler, David (2000), *A First Course in Fourier Analysis*, Prentice Hall, ISBN 0-13-578782-3
- Katznelson, Yitzhak (1976), *An introduction to Harmonic Analysis*, Dover, ISBN 0-486-63331-4
- Knapp, Anthony W. (2001), *Representation Theory of Semisimple Groups: An Overview Based on Examples*^[2], Princeton University Press, ISBN 978-0-691-09089-4
- Pinsky, Mark (2002), *Introduction to Fourier Analysis and Wavelets*, Brooks/Cole, ISBN 0-534-37660-6
- Polyanin, A. D.; Manzhirov, A. V. (1998), *Handbook of Integral Equations*, Boca Raton: CRC Press, ISBN 0-8493-2876-4.
- Rudin, Walter (1987), *Real and Complex Analysis* (Third ed.), Singapore: McGraw Hill, ISBN 0-07-100276-6.

- Stein, Elias; Shakarchi, Rami (2003), *Fourier Analysis: An introduction*, Princeton University Press, ISBN 0-691-11384-X.
- Stein, Elias; Weiss, Guido (1971), *Introduction to Fourier Analysis on Euclidean Spaces*, Princeton, N.J.: Princeton University Press, ISBN 978-0-691-08078-9.
- Wilson, R. G. (1995), *Fourier Series and Optical Transform Techniques in Contemporary Optics*, New York: Wiley, ISBN 0471303577.
- Yosida, K. (1968), *Functional Analysis*, Springer-Verlag, ISBN 3-540-58654-7.

External links

- Fourier Series Applet ^[3] (Tip: drag magnitude or phase dots up or down to change the wave form).
- Stephan Bernsee's FFTlab ^[4] (Java Applet)
- Stanford Video Course on the Fourier Transform ^[5]
- Tables of Integral Transforms ^[6] at EqWorld: The World of Mathematical Equations.
- Weisstein, Eric W., "Fourier Transform ^[7]" from MathWorld.
- Fourier Transform Module by John H. Mathews ^[8]
- The DFT "à Pied": Mastering The Fourier Transform in One Day ^[9] at The DSP Dimension
- An Interactive Flash Tutorial for the Fourier Transform ^[10]

References

- [1] <http://books.google.com/?id=TDQJAAAAIAAJ&printsec=frontcover&dq=Th%C3%A9orie+analytique+de+la+chaleur&q>
 - [2] <http://books.google.com/?id=QCcW1h835pwC>
 - [3] <http://www.westga.edu/~jhasbun/osp/Fourier.htm>
 - [4] <http://www.dspdimension.com/fftlab/>
 - [5] <http://www.academicearth.com/courses/the-fourier-transform-and-its-applications>
 - [6] <http://eqworld.ipmnet.ru/en/auxiliary/aux-inttrans.htm>
 - [7] <http://mathworld.wolfram.com/FourierTransform.html>
 - [8] <http://math.fullerton.edu/mathews/c2003/FourierTransformMod.html>
 - [9] <http://www.dspdimension.com/admin/dft-a-pied/>
 - [10] <http://www.fourier-series.com/f-transform/index.html>
-

Discrete Fourier transform

Fourier transforms
Continuous Fourier transform
Fourier series
Discrete Fourier transform
Discrete-time Fourier transform
Related transforms

In mathematics, the **discrete Fourier transform (DFT)** is a specific kind of Fourier transform, used in Fourier analysis. It transforms one function into another, which is called the *frequency domain* representation, or simply the *DFT*, of the original function (which is often a function in the time domain). But the DFT requires an input function that is *discrete* and whose non-zero values have a limited (*finite*) duration. Such inputs are often created by sampling a continuous function, like a person's voice. Unlike the discrete-time Fourier transform (DTFT), it only evaluates enough frequency components to reconstruct the finite segment that was analyzed. Using the DFT implies that the finite segment that is analyzed is one period of an infinitely extended periodic signal; if this is not actually true, a window function has to be used to reduce the artifacts in the spectrum. For the same reason, the inverse DFT cannot reproduce the entire time domain, unless the input happens to be periodic (forever). Therefore it is often said that the DFT is a transform for Fourier analysis of finite-domain discrete-time functions. The sinusoidal basis functions of the decomposition have the same properties.

The input to the DFT is a finite sequence of real or complex numbers (with more abstract generalizations discussed below), making the DFT ideal for processing information stored in computers. In particular, the DFT is widely employed in signal processing and related fields to analyze the frequencies contained in a sampled signal, to solve partial differential equations, and to perform other operations such as convolutions or multiplying large integers. A key enabling factor for these applications is the fact that the DFT can be computed efficiently in practice using a fast Fourier transform (FFT) algorithm.

FFT algorithms are so commonly employed to compute DFTs that the term "FFT" is often used to mean "DFT" in colloquial settings. Formally, there is a clear distinction: "DFT" refers to a mathematical transformation or function, regardless of how it is computed, whereas "FFT" refers to a specific family of algorithms for computing DFTs. The terminology is further blurred by the (now rare) synonym finite Fourier transform for the DFT, which apparently predates the term "fast Fourier transform" (Cooley et al., 1969) but has the same initialism.

Definition

The sequence of N complex numbers $x_0, \dots, x_{N-1}$ is transformed into the sequence of N complex numbers $X_0, \dots, X_{N-1}$ by the DFT according to the formula:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} \quad k = 0, \dots, N-1$$

where i is the imaginary unit and $e^{\frac{2\pi i}{N}}$ is a primitive N 'th root of unity. (This expression can also be written in terms of a DFT matrix; when scaled appropriately it becomes a unitary matrix and the X_k can thus be viewed as coefficients of x in an orthonormal basis.)

The transform is sometimes denoted by the symbol $\mathcal{F}$, as in $\mathbf{X} = \mathcal{F}\{\mathbf{x}\}$ or $\mathcal{F}(\mathbf{x})$ or $\mathcal{F}_{\mathbf{x}}$.

The **inverse discrete Fourier transform (IDFT)** is given by

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{\frac{2\pi i}{N} kn} \quad n = 0, \dots, N-1.$$

A simple description of these equations is that the complex numbers X_k represent the amplitude and phase of the different sinusoidal components of the input "signal" x_n . The DFT computes the X_k from the x_n , while the IDFT shows how to compute the x_n as a sum of sinusoidal components $(1/N)X_k e^{\frac{2\pi i}{N} kn}$ with frequency k/N cycles per sample. By writing the equations in this form, we are making extensive use of Euler's formula to express sinusoids in terms of complex exponentials, which are much easier to manipulate. In the same way, by writing X_k in polar form, we obtain the sinusoid amplitude A_k/N and phase ϕ_k from the complex modulus and argument of X_k , respectively:

$$A_k = |X_k| = \sqrt{\operatorname{Re}(X_k)^2 + \operatorname{Im}(X_k)^2},$$

$$\varphi_k = \arg(X_k) = \operatorname{atan2}(\operatorname{Im}(X_k), \operatorname{Re}(X_k)),$$

where $\operatorname{atan2}$ is the two-argument form of the arctan function. Note that the normalization factor multiplying the DFT and IDFT (here 1 and $1/N$) and the signs of the exponents are merely conventions, and differ in some treatments. The only requirements of these conventions are that the DFT and IDFT have opposite-sign exponents and that the product of their normalization factors be $1/N$. A normalization of $1/\sqrt{N}$ for both the DFT and IDFT makes the transforms unitary, which has some theoretical advantages, but it is often more practical in numerical computation to perform the scaling all at once as above (and a unit scaling can be convenient in other ways).

(The convention of a negative sign in the exponent is often convenient because it means that X_k is the amplitude of a "positive frequency" $2\pi k/N$. Equivalently, the DFT is often thought of as a matched filter: when looking for a frequency of +1, one correlates the incoming signal with a frequency of -1.)

In the following discussion the terms "sequence" and "vector" will be considered interchangeable.

Properties

Completeness

The discrete Fourier transform is an invertible, linear transformation

$$\mathcal{F} : \mathbb{C}^N \rightarrow \mathbb{C}^N$$

with $\mathbb{C}$ denoting the set of complex numbers. In other words, for any $N > 0$, an N -dimensional complex vector has a DFT and an IDFT which are in turn N -dimensional complex vectors.

Orthogonality

The vectors $e^{\frac{2\pi i}{N} kn}$ form an orthogonal basis over the set of N -dimensional complex vectors:

$$\sum_{n=0}^{N-1} \left(e^{\frac{2\pi i}{N} kn} \right) \left(e^{-\frac{2\pi i}{N} k'n} \right) = N \delta_{kk'}$$

where $\delta_{kk'}$ is the Kronecker delta. This orthogonality condition can be used to derive the formula for the IDFT from the definition of the DFT, and is equivalent to the unitarity property below.

The Plancherel theorem and Parseval's theorem

If X_k and Y_k are the DFTs of x_n and y_n respectively then the Plancherel theorem states:

$$\sum_{n=0}^{N-1} x_n y_n^* = \frac{1}{N} \sum_{k=0}^{N-1} X_k Y_k^*$$

where the star denotes complex conjugation. Parseval's theorem is a special case of the Plancherel theorem and states:

$$\sum_{n=0}^{N-1} |x_n|^2 = \frac{1}{N} \sum_{k=0}^{N-1} |X_k|^2.$$

These theorems are also equivalent to the unitary condition below.

Periodicity

If the expression that defines the DFT is evaluated for all integers k instead of just for $k = 0, \dots, N-1$, then the resulting infinite sequence is a periodic extension of the DFT, periodic with period N .

The periodicity can be shown directly from the definition:

$$X_{k+N} \stackrel{\text{def}}{=} \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N}(k+N)n} = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N}kn} \underbrace{e^{-2\pi i n}}_1 = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N}kn} = X_k.$$

Similarly, it can be shown that the IDFT formula leads to a periodic extension.

The shift theorem

Multiplying x_n by a *linear phase* $e^{\frac{2\pi i}{N}nm}$ for some integer m corresponds to a *circular shift* of the output X_k : X_k is replaced by X_{k-m} , where the subscript is interpreted modulo N (i.e., periodically). Similarly, a circular shift of the input x_n corresponds to multiplying the output X_k by a linear phase. Mathematically, if $\{x_n\}$ represents the vector $\mathbf{x}$ then

$$\text{if } \mathcal{F}(\{x_n\})_k = X_k$$

$$\text{then } \mathcal{F}(\{x_n \cdot e^{\frac{2\pi i}{N}nm}\})_k = X_{k-m}$$

$$\text{and } \mathcal{F}(\{x_{n-m}\})_k = X_k \cdot e^{-\frac{2\pi i}{N}km}$$

Circular convolution theorem and cross-correlation theorem

The convolution theorem for the continuous and discrete time Fourier transforms indicates that a convolution of two infinite sequences can be obtained as the inverse transform of the product of the individual transforms. With sequences and transforms of length N , a circularity arises:

$$\begin{aligned} \mathcal{F}^{-1}\{\mathbf{X} \cdot \mathbf{Y}\}_n &\stackrel{\text{def}}{=} \frac{1}{N} \sum_{k=0}^{N-1} X_k \cdot Y_k \cdot e^{\frac{2\pi i}{N}kn} \\ &= \frac{1}{N} \sum_{k=0}^{N-1} \left(\sum_{l=0}^{N-1} x_l e^{-\frac{2\pi i}{N}kl} \right) \cdot \left(\sum_{m=0}^{N-1} y_m e^{-\frac{2\pi i}{N}km} \right) \cdot e^{\frac{2\pi i}{N}kn} \\ &= \sum_{l=0}^{N-1} x_l \sum_{m=0}^{N-1} y_m \left(\frac{1}{N} \sum_{k=0}^{N-1} e^{\frac{2\pi i}{N}k(n-l-m)} \right). \end{aligned}$$

The quantity in parentheses is 0 for all values of m except those of the form $n-l-pN$, where p is any integer.

At those values, it is 1. It can therefore be replaced by an infinite sum of Kronecker delta functions, and we continue accordingly. Note that we can also extend the limits of m to infinity, with the understanding that the x and y sequences are defined as 0 outside $[0, N-1]$:

$$\begin{aligned}
\mathcal{F}^{-1}\{\mathbf{X} \cdot \mathbf{Y}\}_n &= \sum_{l=0}^{N-1} x_l \sum_{m=-\infty}^{\infty} y_m \left(\sum_{p=-\infty}^{\infty} \delta_{m,(n-l-pN)} \right) \\
&= \sum_{l=0}^{N-1} x_l \sum_{p=-\infty}^{\infty} \underbrace{\left(\sum_{m=-\infty}^{\infty} y_m \cdot \delta_{m,(n-l-pN)} \right)}_{y_{n-l-pN}} \\
&= \sum_{l=0}^{N-1} x_l \left(\sum_{p=-\infty}^{\infty} y_{n-l-pN} \right) \stackrel{\text{def}}{=} (\mathbf{x} * \mathbf{y}_N)_n,
\end{aligned}$$

which is the convolution of the $\mathbf{x}$ sequence with a periodically extended $\mathbf{y}$ sequence defined by:

$$(\mathbf{y}_N)_n \stackrel{\text{def}}{=} \sum_{p=-\infty}^{\infty} y_{(n-pN)}.$$

Similarly, it can be shown that:

$$\mathcal{F}^{-1}\{\mathbf{X}^* \cdot \mathbf{Y}\}_n = \sum_{l=0}^{N-1} x_l^* \cdot (y_N)_{n+l} \stackrel{\text{def}}{=} (\mathbf{x} \star \mathbf{y}_N)_n,$$

which is the cross-correlation of $\mathbf{x}$ and $\mathbf{y}_N$.

A direct evaluation of the convolution or correlation summation (above) requires $O(N^2)$ operations for an output sequence of length N . An indirect method, using transforms, can take advantage of the $O(N \log N)$ efficiency of the fast Fourier transform (FFT) to achieve much better performance. Furthermore, convolutions can be used to efficiently compute DFTs via Rader's FFT algorithm and Bluestein's FFT algorithm. Methods have also been developed to use circular convolution as part of an efficient process that achieves normal (non-circular) convolution with an $\mathbf{x}$ or $\mathbf{y}$ sequence potentially much longer than the practical transform size (N). Two such methods are called overlap-save and overlap-add^[1].

Convolution theorem duality

It can also be shown that:

$$\begin{aligned}
\mathcal{F}\{\mathbf{x} \cdot \mathbf{y}\}_k &\stackrel{\text{def}}{=} \sum_{n=0}^{N-1} x_n \cdot y_n \cdot e^{-\frac{2\pi i}{N} kn} \\
&= \frac{1}{N} (\mathbf{X} * \mathbf{Y}_N)_k, \text{ which is the circular convolution of } \mathbf{X} \text{ and } \mathbf{Y}.
\end{aligned}$$

Trigonometric interpolation polynomial

The trigonometric interpolation polynomial

$$\begin{aligned}
p(t) &= \frac{1}{N} \left[X_0 + X_1 e^{it} + \cdots + X_{N/2-1} e^{(N/2-1)it} + X_{N/2} \cos(Nt/2) + X_{N/2+1} e^{(-N/2+1)it} + \cdots + X_{N-1} e^{-it} \right] \\
&\text{N even,} \\
p(t) &= \frac{1}{N} \left[X_0 + X_1 e^{it} + \cdots + X_{\lfloor N/2 \rfloor} e^{\lfloor N/2 \rfloor it} + X_{\lfloor N/2 \rfloor + 1} e^{-\lfloor N/2 \rfloor it} + \cdots + X_{N-1} e^{-it} \right] \text{ for } N \\
&\text{odd,}
\end{aligned}$$

where the coefficients X_k are given by the DFT of x_n above, satisfies the interpolation property $p(2\pi n/N) = x_n$ for $n = 0, \dots, N-1$.

For even N , notice that the Nyquist component $\frac{X_{N/2}}{N} \cos(Nt/2)$ is handled specially.

This interpolation is *not unique*: aliasing implies that one could add N to any of the complex-sinusoid frequencies (e.g. changing e^{-it} to $e^{i(N-1)t}$) without changing the interpolation property, but giving *different* values in between the x_n points. The choice above, however, is typical because it has two useful properties. First, it consists of sinusoids whose frequencies have the smallest possible magnitudes, and therefore minimizes the mean-square slope $\int |p'(t)|^2 dt$ of the interpolating function. Second, if the x_n are real numbers, then $p(t)$ is real as well.

In contrast, the most obvious trigonometric interpolation polynomial is the one in which the frequencies range from 0 to $N-1$ (instead of roughly $-N/2$ to $+N/2$ as above), similar to the inverse DFT formula. This interpolation does *not* minimize the slope, and is *not* generally real-valued for real x_n ; its use is a common mistake.

The unitary DFT

Another way of looking at the DFT is to note that in the above discussion, the DFT can be expressed as a Vandermonde matrix:

$$\mathbf{F} = \begin{bmatrix} \omega_N^{0 \cdot 0} & \omega_N^{0 \cdot 1} & \dots & \omega_N^{0 \cdot (N-1)} \\ \omega_N^{1 \cdot 0} & \omega_N^{1 \cdot 1} & \dots & \omega_N^{1 \cdot (N-1)} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_N^{(N-1) \cdot 0} & \omega_N^{(N-1) \cdot 1} & \dots & \omega_N^{(N-1) \cdot (N-1)} \end{bmatrix}$$

where

$$\omega_N = e^{-2\pi i/N}$$

is a primitive N th root of unity. The inverse transform is then given by the inverse of the above matrix:

$$\mathbf{F}^{-1} = \frac{1}{N} \mathbf{F}^*$$

With unitary normalization constants $1/\sqrt{N}$, the DFT becomes a unitary transformation, defined by a unitary matrix:

$$\mathbf{U} = \mathbf{F}/\sqrt{N}$$

$$\mathbf{U}^{-1} = \mathbf{U}^*$$

$$|\det(\mathbf{U})| = 1$$

where $\det()$ is the determinant function. The determinant is the product of the eigenvalues, which are always ± 1 or $\pm i$ as described below. In a real vector space, a unitary transformation can be thought of as simply a rigid rotation of the coordinate system, and all of the properties of a rigid rotation can be found in the unitary DFT.

The orthogonality of the DFT is now expressed as an orthonormality condition (which arises in many areas of mathematics as described in root of unity):

$$\sum_{m=0}^{N-1} U_{km} U_{mn}^* = \delta_{kn}$$

If $\mathbf{X}$ is defined as the unitary DFT of the vector $\mathbf{x}$ then

$$X_k = \sum_{n=0}^{N-1} U_{kn} x_n$$

and the Plancherel theorem is expressed as:

$$\sum_{n=0}^{N-1} x_n y_n^* = \sum_{k=0}^{N-1} X_k Y_k^*$$

If we view the DFT as just a coordinate transformation which simply specifies the components of a vector in a new coordinate system, then the above is just the statement that the dot product of two vectors is preserved under a unitary DFT transformation. For the special case $\mathbf{X} = \mathbf{Y}$, this implies that the length of a vector is preserved as

well—this is just Parseval's theorem:

$$\sum_{n=0}^{N-1} |x_n|^2 = \sum_{k=0}^{N-1} |X_k|^2$$

Expressing the inverse DFT in terms of the DFT

A useful property of the DFT is that the inverse DFT can be easily expressed in terms of the (forward) DFT, via several well-known "tricks". (For example, in computations, it is often convenient to only implement a fast Fourier transform corresponding to one transform direction and then to get the other transform direction from the first.)

First, we can compute the inverse DFT by reversing the inputs:

$$\mathcal{F}^{-1}(\{x_n\}) = \mathcal{F}(\{x_{N-n}\})/N$$

(As usual, the subscripts are interpreted modulo N ; thus, for $n = 0$, we have $x_{N-0} = x_0$.)

Second, one can also conjugate the inputs and outputs:

$$\mathcal{F}^{-1}(\mathbf{x}) = \mathcal{F}(\mathbf{x}^*)^*/N$$

Third, a variant of this conjugation trick, which is sometimes preferable because it requires no modification of the data values, involves swapping real and imaginary parts (which can be done on a computer simply by modifying pointers). Define $\text{swap}(x_n)$ as x_n with its real and imaginary parts swapped—that is, if $x_n = a + bi$ then $\text{swap}(x_n)$ is $b + ai$. Equivalently, $\text{swap}(x_n)$ equals ix_n^* . Then

$$\mathcal{F}^{-1}(\mathbf{x}) = \text{swap}(\mathcal{F}(\text{swap}(\mathbf{x}))) / N$$

That is, the inverse transform is the same as the forward transform with the real and imaginary parts swapped for both input and output, up to a normalization (Duhamel *et al.*, 1988).

The conjugation trick can also be used to define a new transform, closely related to the DFT, that is involutory—that is, which is its own inverse. In particular, $T(\mathbf{x}) = \mathcal{F}(\mathbf{x}^*)/\sqrt{N}$ is clearly its own inverse: $T(T(\mathbf{x})) = \mathbf{x}$. A closely related involutory transformation (by a factor of $(1+i)^{1/2}$) is $H(\mathbf{x}) = \mathcal{F}((1+i)\mathbf{x}^*)/\sqrt{2N}$, since the $(1+i)$ factors in $H(H(\mathbf{x}))$ cancel the 2. For real inputs $\mathbf{x}$, the real part of $H(\mathbf{x})$ is none other than the discrete Hartley transform, which is also involutory.

Eigenvalues and eigenvectors

The eigenvalues of the DFT matrix are simple and well-known, whereas the eigenvectors are complicated, not unique, and are the subject of ongoing research.

Consider the unitary form $\mathbf{U}$ defined above for the DFT of length N , where

$$\mathbf{U}_{m,n} = \frac{1}{\sqrt{N}} \omega_N^{(m-1)(n-1)} = \frac{1}{\sqrt{N}} e^{-\frac{2\pi i}{N}(m-1)(n-1)}.$$

This matrix satisfies the equation:

$$\mathbf{U}^4 = \mathbf{I}.$$

This can be seen from the inverse properties above: operating $\mathbf{U}$ twice gives the original data in reverse order, so operating $\mathbf{U}$ four times gives back the original data and is thus the identity matrix. This means that the eigenvalues λ satisfy a characteristic equation:

$$\lambda^4 = 1.$$

Therefore, the eigenvalues of $\mathbf{U}$ are the fourth roots of unity: λ is $+1$, -1 , $+i$, or $-i$.

Since there are only four distinct eigenvalues for this $N \times N$ matrix, they have some multiplicity. The multiplicity gives the number of linearly independent eigenvectors corresponding to each eigenvalue. (Note that there are N independent eigenvectors; a unitary matrix is never defective.)

The problem of their multiplicity was solved by McClellan and Parks (1972), although it was later shown to have been equivalent to a problem solved by Gauss (Dickinson and Steiglitz, 1982). The multiplicity depends on the value of N modulo 4, and is given by the following table:

Multiplicities of the eigenvalues λ of the unitary DFT matrix U as a function of the transform size N (in terms of an integer m).

size N	$\lambda = +1$	$\lambda = -1$	$\lambda = -i$	$\lambda = +i$
$4m$	$m + 1$	m	m	$m - 1$
$4m + 1$	$m + 1$	m	m	m
$4m + 2$	$m + 1$	$m + 1$	m	m
$4m + 3$	$m + 1$	$m + 1$	$m + 1$	m

No simple analytical formula for general eigenvectors is known. Moreover, the eigenvectors are not unique because any linear combination of eigenvectors for the same eigenvalue is also an eigenvector for that eigenvalue. Various researchers have proposed different choices of eigenvectors, selected to satisfy useful properties like orthogonality and to have "simple" forms (e.g., McClellan and Parks, 1972; Dickinson and Steiglitz, 1982; Grünbaum, 1982; Atakishiyev and Wolf, 1997; Candan *et al.*, 2000; Hanna *et al.*, 2004; Gurevich and Hadani, 2008). However two simple closed-form analytical eigenvectors for special DFT period N were found (Kong, 2008):

For DFT period $N = 2L + 1 = 4K + 1$, where K is an integer, the following is an eigenvector of DFT:

$$F(m) = \prod_{s=K+1}^L \left[\cos\left(\frac{2\pi}{N}m\right) - \cos\left(\frac{2\pi}{N}s\right) \right]$$

For DFT period $N = 2L = 4K$, where K is an integer, the following is an eigenvector of DFT:

$$F(m) = \sin\left(\frac{2\pi}{N}m\right) \prod_{s=K+1}^{L-1} \left[\cos\left(\frac{2\pi}{N}m\right) - \cos\left(\frac{2\pi}{N}s\right) \right]$$

The choice of eigenvectors of the DFT matrix has become important in recent years in order to define a discrete analogue of the fractional Fourier transform—the DFT matrix can be taken to fractional powers by exponentiating the eigenvalues (e.g., Rubio and Santhanam, 2005). For the continuous Fourier transform, the natural orthogonal eigenfunctions are the Hermite functions, so various discrete analogues of these have been employed as the eigenvectors of the DFT, such as the Kravchuk polynomials (Atakishiyev and Wolf, 1997). The "best" choice of eigenvectors to define a fractional discrete Fourier transform remains an open question, however.

The real-input DFT

If $x_0, \dots, x_{N-1}$ are real numbers, as they often are in practical applications, then the DFT obeys the symmetry:

$$X_k = X_{N-k}^*.$$

The star denotes complex conjugation. The subscripts are interpreted modulo N .

Therefore, the DFT output for real inputs is half redundant, and one obtains the complete information by only looking at roughly half of the outputs $X_0, \dots, X_{N-1}$. In this case, the "DC" element X_0 is purely real, and for even N the "Nyquist" element $X_{N/2}$ is also real, so there are exactly N non-redundant real numbers in the first half + Nyquist element of the complex output X .

Using Euler's formula, the interpolating trigonometric polynomial can then be interpreted as a sum of sine and cosine functions.

Generalized/shifted DFT

It is possible to shift the transform sampling in time and/or frequency domain by some real shifts a and b , respectively. This is sometimes known as a **generalized DFT** (or **GDFT**), also called the **shifted DFT** or **offset DFT**, and has analogous properties to the ordinary DFT:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N}(k+b)(n+a)} \quad k = 0, \dots, N-1.$$

Most often, shifts of $1/2$ (half a sample) are used. While the ordinary DFT corresponds to a periodic signal in both time and frequency domains, $a = 1/2$ produces a signal that is anti-periodic in frequency domain ($X_{k+N} = -X_k$) and vice-versa for $b = 1/2$. Thus, the specific case of $a = b = 1/2$ is known as an *odd-time odd-frequency* discrete Fourier transform (or O^2 DFT). Such shifted transforms are most often used for symmetric data, to represent different boundary symmetries, and for real-symmetric data they correspond to different forms of the discrete cosine and sine transforms.

Another interesting choice is $a = b = -(N-1)/2$, which is called the **centered DFT** (or **CDFT**). The centered DFT has the useful property that, when N is a multiple of four, all four of its eigenvalues (see above) have equal multiplicities (Rubio and Santhanam, 2005)^[2]

The discrete Fourier transform can be viewed as a special case of the z-transform, evaluated on the unit circle in the complex plane; more general z-transforms correspond to *complex* shifts a and b above.

Multidimensional DFT

The ordinary DFT transforms a one-dimensional sequence or array x_n that is a function of exactly one discrete variable n . The multidimensional DFT of a multidimensional array $x_{n_1, n_2, \dots, n_d}$ that is a function of d discrete variables $n_\ell = 0, 1, \dots, N_\ell - 1$ for ℓ in $1, 2, \dots, d$ is defined by:

$$X_{k_1, k_2, \dots, k_d} = \sum_{n_1=0}^{N_1-1} \left(\omega_{N_1}^{k_1 n_1} \sum_{n_2=0}^{N_2-1} \left(\omega_{N_2}^{k_2 n_2} \dots \sum_{n_d=0}^{N_d-1} \omega_{N_d}^{k_d n_d} \cdot x_{n_1, n_2, \dots, n_d} \right) \dots \right),$$

where $\omega_{N_\ell} = \exp(-2\pi i / N_\ell)$ as above and the d output indices run from $k_\ell = 0, 1, \dots, N_\ell - 1$. This is more compactly expressed in vector notation, where we define $\mathbf{n} = (n_1, n_2, \dots, n_d)$ and $\mathbf{k} = (k_1, k_2, \dots, k_d)$ as d -dimensional vectors of indices from 0 to $\mathbf{N} - 1$, which we define as $\mathbf{N} - 1 = (N_1 - 1, N_2 - 1, \dots, N_d - 1)$:

$$X_{\mathbf{k}} = \sum_{\mathbf{n}=0} e^{-2\pi i \mathbf{k} \cdot (\mathbf{n}/\mathbf{N})} x_{\mathbf{n}},$$

where the division $\mathbf{n}/\mathbf{N}$ is defined as $\mathbf{n}/\mathbf{N} = (n_1/N_1, \dots, n_d/N_d)$ to be performed element-wise, and the sum denotes the set of nested summations above.

The inverse of the multi-dimensional DFT is, analogous to the one-dimensional case, given by:

$$x_{\mathbf{n}} = \frac{1}{\prod_{\ell=1}^d N_\ell} \sum_{\mathbf{k}=0}^{\mathbf{N}-1} e^{2\pi i \mathbf{n} \cdot (\mathbf{k}/\mathbf{N})} X_{\mathbf{k}}.$$

As the one-dimensional DFT expresses the input x_n as a superposition of sinusoids, the multidimensional DFT expresses the input as a superposition of plane waves, or sinusoids. The direction of oscillation in space is $\mathbf{k}/\mathbf{N}$.

The amplitudes are $X_{\mathbf{k}}$. This decomposition is of great importance for everything from digital image processing (two-dimensional) to solving partial differential equations. The solution is broken up into plane waves.

The multidimensional DFT can be computed by the composition of a sequence of one-dimensional DFTs along each dimension. In the two-dimensional case x_{n_1, n_2} the N_1 independent DFTs of the rows (i.e., along n_2) are computed first to form a new array y_{n_1, k_2} . Then the N_2 independent DFTs of y along the columns (along n_1) are

computed to form the final result X_{k_1, k_2} . Alternatively the columns can be computed first and then the rows. The order is immaterial because the nested summations above commute.

An algorithm to compute a one-dimensional DFT is thus sufficient to efficiently compute a multidimensional DFT. This approach is known as the *row-column* algorithm. There are also intrinsically multidimensional FFT algorithms.

The real-input multidimensional DFT

For input data $x_{n_1, n_2, \dots, n_d}$ consisting of real numbers, the DFT outputs have a conjugate symmetry similar to the one-dimensional case above:

$$X_{k_1, k_2, \dots, k_d} = X_{N_1 - k_1, N_2 - k_2, \dots, N_d - k_d}^*$$

where the star again denotes complex conjugation and the ℓ -th subscript is again interpreted modulo N_ℓ (for $\ell = 1, 2, \dots, d$).

Applications

The DFT has seen wide usage across a large number of fields; we only sketch a few examples below (see also the references at the end). All applications of the DFT depend crucially on the availability of a fast algorithm to compute discrete Fourier transforms and their inverses, a fast Fourier transform.

Spectral analysis

When the DFT is used for spectral analysis, the $\{x_n\}$ sequence usually represents a finite set of uniformly-spaced time-samples of some signal $x(t)$, where t represents time. The conversion from continuous time to samples (discrete-time) changes the underlying Fourier transform of $x(t)$ into a discrete-time Fourier transform (DTFT), which generally entails a type of distortion called aliasing. Choice of an appropriate sample-rate (see Nyquist frequency) is the key to minimizing that distortion. Similarly, the conversion from a very long (or infinite) sequence to a manageable size entails a type of distortion called *leakage*, which is manifested as a loss of detail (aka resolution) in the DTFT. Choice of an appropriate sub-sequence length is the primary key to minimizing that effect. When the available data (and time to process it) is more than the amount needed to attain the desired frequency resolution, a standard technique is to perform multiple DFTs, for example to create a spectrogram. If the desired result is a power spectrum and noise or randomness is present in the data, averaging the magnitude components of the multiple DFTs is a useful procedure to reduce the variance of the spectrum (also called a periodogram in this context); two examples of such techniques are the Welch method and the Bartlett method; the general subject of estimating the power spectrum of a noisy signal is called spectral estimation.

A final source of distortion (or perhaps *illusion*) is the DFT itself, because it is just a discrete sampling of the DTFT, which is a function of a continuous frequency domain. That can be mitigated by increasing the resolution of the DFT. That procedure is illustrated in the discrete-time Fourier transform article.

- The procedure is sometimes referred to as *zero-padding*, which is a particular implementation used in conjunction with the fast Fourier transform (FFT) algorithm. The inefficiency of performing multiplications and additions with zero-valued "samples" is more than offset by the inherent efficiency of the FFT.
- As already noted, leakage imposes a limit on the inherent resolution of the DTFT. So there is a practical limit to the benefit that can be obtained from a fine-grained DFT.

Data compression

The field of digital signal processing relies heavily on operations in the frequency domain (i.e. on the Fourier transform). For example, several lossy image and sound compression methods employ the discrete Fourier transform: the signal is cut into short segments, each is transformed, and then the Fourier coefficients of high frequencies, which are assumed to be unnoticeable, are discarded. The decompressor computes the inverse transform based on this reduced number of Fourier coefficients. (Compression applications often use a specialized form of the DFT, the discrete cosine transform or sometimes the modified discrete cosine transform.)

Partial differential equations

Discrete Fourier transforms are often used to solve partial differential equations, where again the DFT is used as an approximation for the Fourier series (which is recovered in the limit of infinite N). The advantage of this approach is that it expands the signal in complex exponentials e^{inx} , which are eigenfunctions of differentiation: $d/dx e^{inx} = in e^{inx}$. Thus, in the Fourier representation, differentiation is simple—we just multiply by in . (Note, however, that the choice of n is not unique due to aliasing; for the method to be convergent, a choice similar to that in the trigonometric interpolation section above should be used.) A linear differential equation with constant coefficients is transformed into an easily solvable algebraic equation. One then uses the inverse DFT to transform the result back into the ordinary spatial representation. Such an approach is called a spectral method.

Polynomial multiplication

Suppose we wish to compute the polynomial product $c(x) = a(x) \cdot b(x)$. The ordinary product expression for the coefficients of c involves a linear (acyclic) convolution, where indices do not "wrap around." This can be rewritten as a cyclic convolution by taking the coefficient vectors for $a(x)$ and $b(x)$ with constant term first, then appending zeros so that the resultant coefficient vectors $\mathbf{a}$ and $\mathbf{b}$ have dimension $d > \deg(a(x)) + \deg(b(x))$. Then,

$$\mathbf{c} = \mathbf{a} * \mathbf{b}$$

Where $\mathbf{c}$ is the vector of coefficients for $c(x)$, and the convolution operator $*$ is defined so

$$c_n = \sum_{m=0}^{d-1} a_m b_{n-m \bmod d} \quad n = 0, 1, \dots, d-1$$

But convolution becomes multiplication under the DFT:

$$\mathcal{F}(\mathbf{c}) = \mathcal{F}(\mathbf{a})\mathcal{F}(\mathbf{b})$$

Here the vector product is taken elementwise. Thus the coefficients of the product polynomial $c(x)$ are just the terms $0, \dots, \deg(a(x)) + \deg(b(x))$ of the coefficient vector

$$\mathbf{c} = \mathcal{F}^{-1}(\mathcal{F}(\mathbf{a})\mathcal{F}(\mathbf{b})).$$

With a fast Fourier transform, the resulting algorithm takes $O(N \log N)$ arithmetic operations. Due to its simplicity and speed, the Cooley–Tukey FFT algorithm, which is limited to composite sizes, is often chosen for the transform operation. In this case, d should be chosen as the smallest integer greater than the sum of the input polynomial degrees that is factorizable into small prime factors (e.g. 2, 3, and 5, depending upon the FFT implementation).

Multiplication of large integers

The fastest known algorithms for the multiplication of very large integers use the polynomial multiplication method outlined above. Integers can be treated as the value of a polynomial evaluated specifically at the number base, with the coefficients of the polynomial corresponding to the digits in that base. After polynomial multiplication, a relatively low-complexity carry-propagation step completes the multiplication.

Some discrete Fourier transform pairs

Some DFT pairs

$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{i2\pi kn/N}$	$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi kn/N}$	Note
$x_n e^{i2\pi n\ell/N}$	$X_{k-\ell}$	Shift theorem
$x_{n-\ell}$	$X_k e^{-i2\pi k\ell/N}$	
$x_n \in \mathbb{R}$	$X_k = X_{N-k}^*$	Real DFT
a^n	$\begin{cases} N & \text{if } a = e^{i2\pi k/N} \\ \frac{1-a^N}{1-a e^{-i2\pi k/N}} & \text{otherwise} \end{cases}$	from the geometric progression formula
$\binom{N-1}{n}$	$(1 + e^{-i2\pi k/N})^{N-1}$	from the binomial theorem
$\begin{cases} \frac{1}{W} & \text{if } 2n < W \text{ or } 2(N-n) < W \\ 0 & \text{otherwise} \end{cases}$	$\begin{cases} 1 & \text{if } k = 0 \\ \frac{\sin(\frac{\pi W k}{N})}{W \sin(\frac{\pi k}{N})} & \text{otherwise} \end{cases}$	x_n is a rectangular window function of W points centered on x_0 , where W is an odd integer, and X_k is a sinc-like function

Derivation as Fourier series

The DFT can be derived as a truncation of the Fourier series of a periodic sequence of impulses.

Generalizations

Representation theory

The DFT can be interpreted as the complex-valued representation theory of the finite cyclic group. In other words, a sequence of n complex numbers can be thought of as an element of n -dimensional complex space $\mathbf{C}^n$, or equivalently a function from the finite cyclic group of order n to the complex numbers, $\mathbf{Z}/n\mathbf{Z} \rightarrow \mathbf{C}$. This latter may be suggestively written $\mathbf{C}^{\mathbf{Z}/n\mathbf{Z}}$ to emphasize that this is a complex vector space whose coordinates are indexed by the n -element set $\mathbf{Z}/n\mathbf{Z}$.

From this point of view, one may generalize the DFT to representation theory generally, or more narrowly to the representation theory of finite groups.

More narrowly still, one may generalize the DFT by either changing the target (taking values in a field other than the complex numbers), or the domain (a group other than a finite cyclic group), as detailed in the sequel.

Other fields

Many of the properties of the DFT only depend on the fact that $e^{-\frac{2\pi i}{N}}$ is a primitive root of unity, sometimes denoted ω_N or W_N (so that $\omega_N^N = 1$). Such properties include the completeness, orthogonality, Plancherel/Parseval, periodicity, shift, convolution, and unitarity properties above, as well as many FFT algorithms. For this reason, the discrete Fourier transform can be defined by using roots of unity in fields other than the complex numbers, and such generalizations are commonly called *number-theoretic transforms* (NTTs) in the case of finite fields. For more information, see number-theoretic transform and discrete Fourier transform (general).

Other finite groups

The standard DFT acts on a sequence $x_0, x_1, \dots, x_{N-1}$ of complex numbers, which can be viewed as a function $\{0, 1, \dots, N-1\} \rightarrow \mathbb{C}$. The multidimensional DFT acts on multidimensional sequences, which can be viewed as functions

$$\{0, 1, \dots, N_1 - 1\} \times \dots \times \{0, 1, \dots, N_d - 1\} \rightarrow \mathbb{C}.$$

This suggests the generalization to Fourier transforms on arbitrary finite groups, which act on functions $G \rightarrow \mathbb{C}$ where G is a finite group. In this framework, the standard DFT is seen as the Fourier transform on a cyclic group, while the multidimensional DFT is a Fourier transform on a direct sum of cyclic groups.

Alternatives

As with other Fourier transforms, there are various alternatives to the DFT for various applications, prominent among which are wavelets. The analog of the DFT is the discrete wavelet transform (DWT). From the point of view of time–frequency analysis, a key limitation of the Fourier transform is that it does not include *location* information, only *frequency* information, and thus has difficulty in representing transients. As wavelets have location as well as frequency, they are better able to represent location, at the expense of greater difficulty representing frequency. For details, see comparison of the discrete wavelet transform with the discrete Fourier transform.

See also

- DFT matrix
- Fast Fourier transform
- List of Fourier-related transforms
- FFTW

References

- Brigham, E. Oran (1988). *The fast Fourier transform and its applications*. Englewood Cliffs, N.J.: Prentice Hall. ISBN 0-13-307505-2.
- Oppenheim, Alan V.; Schafer, R. W.; and Buck, J. R. (1999). *Discrete-time signal processing*. Upper Saddle River, N.J.: Prentice Hall. ISBN 0-13-754920-2.
- Smith, Steven W. (1999). "Chapter 8: The Discrete Fourier Transform" ^[3]. *The Scientist and Engineer's Guide to Digital Signal Processing* (Second ed.). San Diego, Calif.: California Technical Publishing. ISBN 0-9660176-3-3.
- Cormen, Thomas H.; Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein (2001). "Chapter 30: Polynomials and the FFT". *Introduction to Algorithms* (Second ed.). MIT Press and McGraw-Hill. pp. 822–848. ISBN 0-262-03293-7. esp. section 30.2: The DFT and FFT, pp. 830–838.
- P. Duhamel, B. Piron, and J. M. Etcheto (1988). "On computing the inverse DFT". *IEEE Trans. Acoust., Speech and Sig. Processing* **36** (2): 285–286. doi:10.1109/29.1519.
- J. H. McClellan and T. W. Parks (1972). "Eigenvalues and eigenvectors of the discrete Fourier transformation". *IEEE Trans. Audio Electroacoust.* **20** (1): 66–74. doi:10.1109/TAU.1972.1162342.

- Bradley W. Dickinson and Kenneth Steiglitz (1982). "Eigenvectors and functions of the discrete Fourier transform". *IEEE Trans. Acoust., Speech and Sig. Processing* **30** (1): 25–31. doi:10.1109/TASSP.1982.1163843. (Note that this paper has an apparent typo in its table of the eigenvalue multiplicities: the $+i/-i$ columns are interchanged. The correct table can be found in McClellan and Parks, 1972, and is easily confirmed numerically.)
- F. A. Grünbaum (1982). "The eigenvectors of the discrete Fourier transform". *J. Math. Anal. Appl.* **88** (2): 355–363. doi:10.1016/0022-247X(82)90199-8.
- Natig M. Atakishiyev and Kurt Bernardo Wolf (1997). "Fractional Fourier-Kravchuk transform". *J. Opt. Soc. Am. A* **14** (7): 1467–1477. doi:10.1364/JOSAA.14.001467.
- C. Candan, M. A. Kutay and H. M. Ozaktas (2000). "The discrete fractional Fourier transform". *IEEE Trans. on Signal Processing* **48** (5): 1329–1337. doi:10.1109/78.839980.
- Magdy Tawfik Hanna, Nabila Philip Attalla Seif, and Waleed Abd El Maguid Ahmed (2004). "Hermite-Gaussian-like eigenvectors of the discrete Fourier transform matrix based on the singular-value decomposition of its orthogonal projection matrices". *IEEE Trans. Circ. Syst. I* **51** (11): 2245–2254. doi:10.1109/TCSI.2004.836850.
- Shamgar Gurevich and Ronny Hadani (2009). "On the diagonalization of the discrete Fourier transform". *Applied and Computational Harmonic Analysis* **27** (1): 87–99. doi:10.1016/j.acha.2008.11.003. preprint at arXiv:0808.3281.
- Shamgar Gurevich, Ronny Hadani, and Nir Sochen (2008). "The finite harmonic oscillator and its applications to sequences, communication and radar". *IEEE Transactions on Information Theory* **54** (9): 4239–4253. doi:10.1109/TIT.2008.926440. preprint at arXiv:0808.1495.
- Juan G. Vargas-Rubio and Balu Santhanam (2005). "On the multiangle centered discrete fractional Fourier transform". *IEEE Sig. Proc. Lett.* **12** (4): 273–276. doi:10.1109/LSP.2005.843762.
- J. Cooley, P. Lewis, and P. Welch (1969). "The finite Fourier transform". *IEEE Trans. Audio Electroacoustics* **17** (2): 77–85. doi:10.1109/TAU.1969.1162036.
- F.N. Kong (2008). "Analytic Expressions of Two Discrete Hermite-Gaussian Signals". *IEEE Trans. Circuits and Systems –II: Express Briefs.* **55** (1): 56–60. doi:10.1109/TCSII.2007.909865.

External links

- Interactive flash tutorial on the DFT ^[4]
- Mathematics of the Discrete Fourier Transform by Julius O. Smith III ^[5]
- Fast implementation of the DFT - coded in C and under General Public License (GPL) ^[6]
- Example of how DFT spectral analysis is used in engineering studies of the Otto Struve 2.1m telescope ^[7]
- The DFT “à Pied”: Mastering The Fourier Transform in One Day ^[9]

References

- [1] T. G. Stockham, Jr., "High-speed convolution and correlation," in 1966 *Proc. AFIPS Spring Joint Computing Conf.* Reprinted in *Digital Signal Processing*, L. R. Rabiner and C. M. Rader, editors, New York: IEEE Press, 1972.
- [2] Santhanam, Balu; Santhanam, Thalanayar S. "Discrete Gauss-Hermite functions and eigenvectors of the centered discrete Fourier transform" (<http://thamaku.usc.edu/Proceedings/ICASSP 2007/pdfs/0301385.pdf>), Proceedings of the 32nd IEEE *International Conference on Acoustics, Speech, and Signal Processing* (ICASSP 2007, SPTM-P12.4), vol. III, pp. 1385-1388.
- [3] <http://www.dspguide.com/ch8/1.htm>
- [4] http://www.fourier-series.com/fourierseries2/DFT_tutorial.html
- [5] <http://ccrma.stanford.edu/~jos/mdft/mdft.html>
- [6] <http://www.fftw.org>
- [7] http://nexus.as.utexas.edu/kuehne/3_4%22.html

Fast Fourier transform

A **fast Fourier transform (FFT)** is an efficient algorithm to compute the discrete Fourier transform (DFT) and its inverse. There are many distinct FFT algorithms involving a wide range of mathematics, from simple complex-number arithmetic to group theory and number theory; this article gives an overview of the available techniques and some of their general properties, while the specific algorithms are described in subsidiary articles linked below.

A DFT decomposes a sequence of values into components of different frequencies. This operation is useful in many fields (see discrete Fourier transform for properties and applications of the transform) but computing it directly from the definition is often too slow to be practical. An FFT is a way to compute the same result more quickly: computing a DFT of N points in the naive way, using the definition, takes $O(N^2)$ arithmetical operations, while an FFT can compute the same result in only $O(N \log N)$ operations. The difference in speed can be substantial, especially for long data sets where N may be in the thousands or millions—in practice, the computation time can be reduced by several orders of magnitude in such cases, and the improvement is roughly proportional to $N/\log(N)$. This huge improvement made many DFT-based algorithms practical; FFTs are of great importance to a wide variety of applications, from digital signal processing and solving partial differential equations to algorithms for quick multiplication of large integers.

The most well known FFT algorithms depend upon the factorization of N , but (contrary to popular misconception) there are FFTs with $O(N \log N)$ complexity for all N , even for prime N . Many FFT algorithms only depend on the fact that $e^{-\frac{2\pi i}{N}}$ is an N th primitive root of unity, and thus can be applied to analogous transforms over any finite field, such as number-theoretic transforms.

Since the inverse DFT is the same as the DFT, but with the opposite sign in the exponent and a $1/N$ factor, any FFT algorithm can easily be adapted for it.

Definition and speed

An FFT computes the DFT and produces exactly the same result as evaluating the DFT definition directly; the only difference is that an FFT is much faster. (In the presence of round-off error, many FFT algorithms are also much more accurate than evaluating the DFT definition directly, as discussed below.)

Let $x_0, \dots, x_{N-1}$ be complex numbers. The DFT is defined by the formula

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi k \frac{n}{N}} \quad k = 0, \dots, N-1.$$

Evaluating this definition directly requires $O(N^2)$ operations: there are N outputs X_k , and each output requires a sum of N terms. An FFT is any method to compute the same results in $O(N \log N)$ operations. More precisely, all known FFT algorithms require $\Theta(N \log N)$ operations (technically, O only denotes an upper bound), although there is no proof that better complexity is impossible.

To illustrate the savings of an FFT, consider the count of complex multiplications and additions. Evaluating the DFT's sums directly involves N^2 complex multiplications and $N(N-1)$ complex additions [of which $O(N)$ operations can be saved by eliminating trivial operations such as multiplications by 1]. The well-known radix-2 Cooley–Tukey algorithm, for N a power of 2, can compute the same result with only $(N/2) \log_2 N$ complex multiplies (again, ignoring simplifications of multiplications by 1 and similar) and $N \log_2 N$ complex additions. In practice, actual performance on modern computers is usually dominated by factors other than arithmetic and is a complicated subject (see, e.g., Frigo & Johnson, 2005), but the overall improvement from $\Theta(N^2)$ to $\Theta(N \log N)$ remains.

Computational issues

Bounds on complexity and operation counts

A fundamental question of longstanding theoretical interest is to prove lower bounds on the complexity and exact operation counts of fast Fourier transforms, and many open problems remain. It is not even rigorously proved whether DFTs truly require $\Omega(N \log N)$ (i.e., order $N \log N$ or greater) operations, even for the simple case of power of two sizes, although no algorithms with lower complexity are known. In particular, the count of arithmetic operations is usually the focus of such questions, although actual performance on modern-day computers is determined by many other factors such as cache or CPU pipeline optimization.

Following pioneering work by Winograd (1978), a tight $\Theta(N)$ lower bound is known for the number of real multiplications required by an FFT. It can be shown that only $4N - 2 \log_2^2 N - 2 \log_2 N - 4$ irrational real multiplications are required to compute a DFT of power-of-two length $N = 2^m$. Moreover, explicit algorithms that achieve this count are known (Heideman & Burrus, 1986; Duhamel, 1990). Unfortunately, these algorithms require too many additions to be practical, at least on modern computers with hardware multipliers.

A tight lower bound is *not* known on the number of required additions, although lower bounds have been proved under some restrictive assumptions on the algorithms. In 1973, Morgenstern proved an $\Omega(N \log N)$ lower bound on the addition count for algorithms where the multiplicative constants have bounded magnitudes (which is true for most but not all FFT algorithms). Pan (1986) proved an $\Omega(N \log N)$ lower bound assuming a bound on a measure of the FFT algorithm's "asynchronicity", but the generality of this assumption is unclear. For the case of power-of-two N , Papadimitriou (1979) argued that the number $N \log_2 N$ of complex-number additions achieved by Cooley–Tukey algorithms is *optimal* under certain assumptions on the graph of the algorithm (his assumptions imply, among other things, that no additive identities in the roots of unity are exploited). (This argument would imply that at least $2N \log_2 N$ real additions are required, although this is not a tight bound because extra additions are required as part of complex-number multiplications.) Thus far, no published FFT algorithm has achieved fewer than $N \log_2 N$ complex-number additions (or their equivalent) for power-of-two N .

A third problem is to minimize the *total* number of real multiplications and additions, sometimes called the "arithmetic complexity" (although in this context it is the exact count and not the asymptotic complexity that is being considered). Again, no tight lower bound has been proven. Since 1968, however, the lowest published count for power-of-two N was long achieved by the split-radix FFT algorithm, which requires $4N \log_2 N - 6N + 8$ real multiplications and additions for $N > 1$. This was recently reduced to $\sim \frac{34}{9} N \log_2 N$ (Johnson and Frigo, 2007; Lundy and Van Buskirk, 2007).

Most of the attempts to lower or prove the complexity of FFT algorithms have focused on the ordinary complex-data case, because it is the simplest. However, complex-data FFTs are so closely related to algorithms for related problems such as real-data FFTs, discrete cosine transforms, discrete Hartley transforms, and so on, that any improvement in one of these would immediately lead to improvements in the others (Duhamel & Vetterli, 1990).

Accuracy and approximations

All of the FFT algorithms discussed below compute the DFT exactly (in exact arithmetic, i.e. neglecting floating-point errors). A few "FFT" algorithms have been proposed, however, that compute the DFT *approximately*, with an error that can be made arbitrarily small at the expense of increased computations. Such algorithms trade the approximation error for increased speed or other properties. For example, an approximate FFT algorithm by Edelman et al. (1999) achieves lower communication requirements for parallel computing with the help of a fast multipole method. A wavelet-based approximate FFT by Guo and Burrus (1996) takes sparse inputs/outputs (time/frequency localization) into account more efficiently than is possible with an exact FFT. Another algorithm for approximate computation of a subset of the DFT outputs is due to Shentov et al. (1995). Only the Edelman algorithm

works equally well for sparse and non-sparse data, however, since it is based on the compressibility (rank deficiency) of the Fourier matrix itself rather than the compressibility (sparsity) of the data.

Even the "exact" FFT algorithms have errors when finite-precision floating-point arithmetic is used, but these errors are typically quite small; most FFT algorithms, e.g. Cooley–Tukey, have excellent numerical properties. The upper bound on the relative error for the Cooley–Tukey algorithm is $O(\epsilon \log N)$, compared to $O(\epsilon N^{3/2})$ for the naïve DFT formula (Gentleman and Sande, 1966), where ϵ is the machine floating-point relative precision. In fact, the root mean square (rms) errors are much better than these upper bounds, being only $O(\epsilon \sqrt{\log N})$ for Cooley–Tukey and $O(\epsilon \sqrt{N})$ for the naïve DFT (Schatzman, 1996). These results, however, are very sensitive to the accuracy of the twiddle factors used in the FFT (i.e. the trigonometric function values), and it is not unusual for incautious FFT implementations to have much worse accuracy, e.g. if they use inaccurate trigonometric recurrence formulas. Some FFTs other than Cooley–Tukey, such as the Rader-Brenner algorithm, are intrinsically less stable.

In fixed-point arithmetic, the finite-precision errors accumulated by FFT algorithms are worse, with rms errors growing as $O(\sqrt{N})$ for the Cooley–Tukey algorithm (Welch, 1969). Moreover, even achieving this accuracy requires careful attention to scaling in order to minimize the loss of precision, and fixed-point FFT algorithms involve rescaling at each intermediate stage of decompositions like Cooley–Tukey.

To verify the correctness of an FFT implementation, rigorous guarantees can be obtained in $O(N \log N)$ time by a simple procedure checking the linearity, impulse-response, and time-shift properties of the transform on random inputs (Ergün, 1995).

Algorithms

Cooley–Tukey algorithm

By far the most common FFT is the Cooley–Tukey algorithm. This is a divide and conquer algorithm that recursively breaks down a DFT of any composite size $N = N_1 N_2$ into many smaller DFTs of sizes N_1 and N_2 , along with $O(N)$ multiplications by complex roots of unity traditionally called twiddle factors (after Gentleman and Sande, 1966).

This method (and the general idea of an FFT) was popularized by a publication of J. W. Cooley and J. W. Tukey in 1965, but it was later discovered (Heideman & Burrus, 1984) that those two authors had independently re-invented an algorithm known to Carl Friedrich Gauss around 1805 (and subsequently rediscovered several times in limited forms).

The most well-known use of the Cooley–Tukey algorithm is to divide the transform into two pieces of size $N/2$ at each step, and is therefore limited to power-of-two sizes, but any factorization can be used in general (as was known to both Gauss and Cooley/Tukey). These are called the **radix-2** and **mixed-radix** cases, respectively (and other variants such as the split-radix FFT have their own names as well). Although the basic idea is recursive, most traditional implementations rearrange the algorithm to avoid explicit recursion. Also, because the Cooley–Tukey algorithm breaks the DFT into smaller DFTs, it can be combined arbitrarily with any other algorithm for the DFT, such as those described below.

Other FFT algorithms

There are other FFT algorithms distinct from Cooley–Tukey. For $N = N_1 N_2$ with coprime N_1 and N_2 , one can use the Prime-Factor (Good-Thomas) algorithm (PFA), based on the Chinese Remainder Theorem, to factorize the DFT similarly to Cooley–Tukey but without the twiddle factors. The Rader-Brenner algorithm (1976) is a Cooley–Tukey-like factorization but with purely imaginary twiddle factors, reducing multiplications at the cost of increased additions and reduced numerical stability; it was later superseded by the split-radix variant of Cooley–Tukey (which achieves the same multiplication count but with fewer additions and without sacrificing accuracy). Algorithms that recursively factorize the DFT into smaller operations other than DFTs include the Bruun

and QFT algorithms. (The Rader-Brenner and QFT algorithms were proposed for power-of-two sizes, but it is possible that they could be adapted to general composite n . Bruun's algorithm applies to arbitrary even composite sizes.) Bruun's algorithm, in particular, is based on interpreting the FFT as a recursive factorization of the polynomial $z^N - 1$, here into real-coefficient polynomials of the form $z^M - 1$ and $z^{2M} + az^M + 1$.

Another polynomial viewpoint is exploited by the Winograd algorithm, which factorizes $z^N - 1$ into cyclotomic polynomials—these often have coefficients of 1, 0, or -1 , and therefore require few (if any) multiplications, so Winograd can be used to obtain minimal-multiplication FFTs and is often used to find efficient algorithms for small factors. Indeed, Winograd showed that the DFT can be computed with only $O(N)$ irrational multiplications, leading to a proven achievable lower bound on the number of multiplications for power-of-two sizes; unfortunately, this comes at the cost of many more additions, a tradeoff no longer favorable on modern processors with hardware multipliers. In particular, Winograd also makes use of the PFA as well as an algorithm by Rader for FFTs of *prime* sizes.

Rader's algorithm, exploiting the existence of a generator for the multiplicative group modulo prime N , expresses a DFT of prime size n as a cyclic convolution of (composite) size $N - 1$, which can then be computed by a pair of ordinary FFTs via the convolution theorem (although Winograd uses other convolution methods). Another prime-size FFT is due to L. I. Bluestein, and is sometimes called the chirp-z algorithm; it also re-expresses a DFT as a convolution, but this time of the *same* size (which can be zero-padded to a power of two and evaluated by radix-2 Cooley–Tukey FFTs, for example), via the identity $nk = -(k - n)^2/2 + n^2/2 + k^2/2$.

FFT algorithms specialized for real and/or symmetric data

In many applications, the input data for the DFT are purely real, in which case the outputs satisfy the symmetry

$$X_{N-k} = X_k^*,$$

and efficient FFT algorithms have been designed for this situation (see e.g. Sorensen, 1987). One approach consists of taking an ordinary algorithm (e.g. Cooley–Tukey) and removing the redundant parts of the computation, saving roughly a factor of two in time and memory. Alternatively, it is possible to express an *even*-length real-input DFT as a complex DFT of half the length (whose real and imaginary parts are the even/odd elements of the original real data), followed by $O(N)$ post-processing operations.

It was once believed that real-input DFTs could be more efficiently computed by means of the discrete Hartley transform (DHT), but it was subsequently argued that a specialized real-input DFT algorithm (FFT) can typically be found that requires fewer operations than the corresponding DHT algorithm (FHT) for the same number of inputs. Bruun's algorithm (above) is another method that was initially proposed to take advantage of real inputs, but it has not proved popular.

There are further FFT specializations for the cases of real data that have even/odd symmetry, in which case one can gain another factor of (roughly) two in time and memory and the DFT becomes the discrete cosine/sine transform(s) (DCT/DST). Instead of directly modifying an FFT algorithm for these cases, DCTs/DSTs can also be computed via FFTs of real data combined with $O(N)$ pre/post processing.

Multidimensional FFTs

As defined in the multidimensional DFT article, the multidimensional DFT

$$X_{\mathbf{k}} = \sum_{\mathbf{n}=0}^{N-1} e^{-2\pi i \mathbf{k} \cdot (\mathbf{n}/N)} x_{\mathbf{n}}$$

transforms an array $x_{\mathbf{n}}$ with a d -dimensional vector of indices $\mathbf{n} = (n_1, n_2, \dots, n_d)$ by a set of d nested summations (over $n_j = 0 \dots N_j - 1$ for each j), where the division $\mathbf{n}/N$, defined as $\mathbf{n}/N = (n_1/N_1, \dots, n_d/N_d)$, is performed element-wise. Equivalently, it is simply the composition of a sequence of d sets of one-dimensional DFTs, performed along one dimension at a time (in any order). This compositional viewpoint immediately provides the simplest and most common multidimensional DFT algorithm, known as the **row-column** algorithm (after the two-dimensional case, below). That is, one simply performs a sequence of d one-dimensional FFTs (by any of the above algorithms): first you transform along the n_1 dimension, then along the n_2 dimension, and so on (or actually, any ordering will work). This method is easily shown to have the usual $O(N \log N)$ complexity, where $N = N_1 N_2 \dots N_d$ is the total number of data points transformed. In particular, there are N/N_1 transforms of size N_1 , etcetera, so the complexity of the sequence of FFTs is:

$$\begin{aligned} & \frac{N}{N_1} O(N_1 \log N_1) + \dots + \frac{N}{N_d} O(N_d \log N_d) \\ &= O(N [\log N_1 + \dots + \log N_d]) = O(N \log N). \end{aligned}$$

In two dimensions, the $x_{\mathbf{k}}$ can be viewed as an $n_1 \times n_2$ matrix, and this algorithm corresponds to first performing the FFT of all the rows and then of all the columns (or vice versa), hence the name.

In more than two dimensions, it is often advantageous for cache locality to group the dimensions recursively. For example, a three-dimensional FFT might first perform two-dimensional FFTs of each planar "slice" for each fixed n_1 , and then perform the one-dimensional FFTs along the n_1 direction. More generally, an asymptotically optimal cache-oblivious algorithm consists of recursively dividing the dimensions into two groups $(n_1, \dots, n_{d/2})$ and $(n_{d/2+1}, \dots, n_d)$ that are transformed recursively (rounding if d is not even) (see Frigo and Johnson, 2005). Still, this remains a straightforward variation of the row-column algorithm that ultimately requires only a one-dimensional FFT algorithm as the base case, and still has $O(N \log N)$ complexity. Yet another variation is to perform matrix transpositions in between transforming subsequent dimensions, so that the transforms operate on contiguous data; this is especially important for out-of-core and distributed memory situations where accessing non-contiguous data is extremely time-consuming.

There are other multidimensional FFT algorithms that are distinct from the row-column algorithm, although all of them have $O(N \log N)$ complexity. Perhaps the simplest non-row-column FFT is the vector-radix FFT algorithm, which is a generalization of the ordinary Cooley–Tukey algorithm where one divides the transform dimensions by a vector $\mathbf{r} = (r_1, r_2, \dots, r_d)$ of radices at each step. (This may also have cache benefits.) The simplest case of vector-radix is where all of the radices are equal (e.g. vector-radix-2 divides *all* of the dimensions by two), but this is not necessary. Vector radix with only a single non-unit radix at a time, i.e. $\mathbf{r} = (1, \dots, 1, r, 1, \dots, 1)$, is essentially a row-column algorithm. Other, more complicated, methods include polynomial transform algorithms due to Nussbaumer (1977), which view the transform in terms of convolutions and polynomial products. See Duhamel and Vetterli (1990) for more information and references.

Other generalizations

An $O(N^{5/2} \log N)$ generalization to spherical harmonics on the sphere S^2 with N^2 nodes was described by Mohlenkamp (1999), along with an algorithm conjectured (but not proven) to have $O(N^2 \log^2 N)$ complexity; Mohlenkamp also provides an implementation in the libftsh library ^[1]. A spherical-harmonic algorithm with $O(N^2 \log N)$ complexity is described by Rokhlin and Tygert (2006).

Various groups have also published "FFT" algorithms for non-equispaced data, as reviewed in Potts *et al.* (2001). Such algorithms do not strictly compute the DFT (which is only defined for equispaced data), but rather some approximation thereof (a non-uniform discrete Fourier transform, or NDFT, which itself is often computed only approximately).

See also

- Split-radix FFT algorithm
- Prime-factor FFT algorithm
- Bruun's FFT algorithm
- Rader's FFT algorithm
- Bluestein's FFT algorithm
- Butterfly diagram - a diagram used to describe FFTs.
- Odlyzko–Schönhage algorithm applies the FFT to finite Dirichlet series.
- Overlap add/Overlap save - efficient convolution methods using FFT for long signals
- Spectral music (involves application of FFT analysis to musical composition)
- Spectrum analyzers - Devices that perform an FFT
- FFTW "Fastest Fourier Transform in the West" - 'C' library for the discrete Fourier transform (DFT) in one or more dimensions.
- Time Series
- Math Kernel Library

References

- N. Brenner and C. Rader, 1976, A New Principle for Fast Fourier Transformation ^[2], *IEEE Acoustics, Speech & Signal Processing* **24**: 264-266.
- Brigham, E.O. (2002), *The Fast Fourier Transform*, New York: Prentice-Hall
- Cooley, James W., and John W. Tukey, 1965, "An algorithm for the machine calculation of complex Fourier series," *Math. Comput.* **19**: 297–301.
- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, 2001. *Introduction to Algorithms*, 2nd. ed. MIT Press and McGraw-Hill. ISBN 0-262-03293-7. Especially chapter 30, "Polynomials and the FFT."
- Pierre Duhamel, 1990, Algorithms meeting the lower bounds on the multiplicative complexity of length- 2^n DFTs and their connection with practical algorithms (doi:10.1109/29.60070), *IEEE Trans. Acoust. Speech. Sig. Proc.* **38**: 1504-151.
- P. Duhamel and M. Vetterli, 1990, Fast Fourier transforms: a tutorial review and a state of the art (doi:10.1016/0165-1684(90)90158-U), *Signal Processing* **19**: 259–299.
- A. Edelman, P. McCorquodale, and S. Toledo, 1999, The Future Fast Fourier Transform? (doi:10.1137/S1064827597316266), *SIAM J. Sci. Computing* **20**: 1094–1114.
- Funda Ergün, 1995, Testing multivariate linear functions: Overcoming the generator bottleneck (doi:10.1145/225058.225167), *Proc. 27th ACM Symposium on the Theory of Computing*: 407–416.
- M. Frigo and S. G. Johnson, 2005, "The Design and Implementation of FFTW3 ^[3]," *Proceedings of the IEEE* **93**: 216–231.

- Carl Friedrich Gauss, 1866. "Nachlass: Theoria interpolationis methodo nova tractata," *Werke* band 3, 265–327. Göttingen: Königliche Gesellschaft der Wissenschaften.
 - W. M. Gentleman and G. Sande, 1966, "Fast Fourier transforms—for fun and profit," *Proc. AFIPS* **29**: 563–578.
 - H. Guo and C. S. Burrus, 1996, Fast approximate Fourier transform via wavelets transform (doi:10.1117/12.255236), *Proc. SPIE Intl. Soc. Opt. Eng.* **2825**: 250–259.
 - H. Guo, G. A. Sittou, C. S. Burrus, 1994, The Quick Discrete Fourier Transform (doi:10.1109/ICASSP.1994.389994), *Proc. IEEE Conf. Acoust. Speech and Sig. Processing (ICASSP)* **3**: 445–448.
 - Heideman, M. T., D. H. Johnson, and C. S. Burrus, "Gauss and the history of the fast Fourier transform ^[4]," *IEEE ASSP Magazine*, 1, (4), 14–21 (1984).
 - Michael T. Heideman and C. Sidney Burrus, 1986, On the number of multiplications necessary to compute a length- 2^n DFT ^[5], *IEEE Trans. Acoust. Speech. Sig. Proc.* **34**: 91–95.
 - S. G. Johnson and M. Frigo, 2007. "A modified split-radix FFT with fewer arithmetic operations ^[6]," *IEEE Trans. Signal Processing* **55** (1): 111–119.
 - T. Lundy and J. Van Buskirk, 2007. "A new matrix approach to real FFTs and convolutions of length 2^k ," *Computing* **80** (1): 23–45.
 - Jacques Morgenstern, 1973, Note on a lower bound of the linear complexity of the fast Fourier transform (doi:10.1145/321752.321761), *J. ACM* **20**: 305–306.
 - M. J. Mohlenkamp, 1999, "A fast transform for spherical harmonics", *J. Fourier Anal. Appl.* **5**, 159–184. ^[7] (preprint ^[8])
 - H. J. Nussbaumer, 1977, Digital filtering using polynomial transforms (doi:10.1049/el:19770280), *Electronics Lett.* **13**: 386–387.
 - V. Pan, 1986, The trade-off between the additive complexity and the asynchronicity of linear and bilinear algorithms (doi:10.1016/0020-0190(86)90035-9), *Information Proc. Lett.* **22**: 11–14.
 - Christos H. Papadimitriou, 1979, Optimality of the fast Fourier transform (doi:10.1145/322108.322118), *J. ACM* **26**: 95–102.
 - D. Potts, G. Steidl, and M. Tasche, 2001. "Fast Fourier transforms for nonequispaced data: A tutorial ^[9]", in: J.J. Benedetto and P. Ferreira (Eds.), *Modern Sampling Theory: Mathematics and Applications* (Birkhauser).
 - Vladimir Rokhlin and Mark Tygert, 2006, "Fast algorithms for spherical harmonic expansions ^[10]," *SIAM J. Sci. Computing* **27** (6): 1903–1928.
 - James C. Schatzman, 1996, Accuracy of the discrete Fourier transform and the fast Fourier transform ^[11], *SIAM J. Sci. Comput.* **17**: 1150–1166.
 - O. V. Shentov, S. K. Mitra, U. Heute, and A. N. Hossen, 1995, Subband DFT. I. Definition, interpretations and extensions (doi:10.1016/0165-1684(94)00103-7), *Signal Processing* **41**: 261–277.
 - H. V. Sorensen, D. L. Jones, M. T. Heideman, and C. S. Burrus, 1987, Real-valued fast Fourier transform algorithms ^[12], *IEEE Trans. Acoust. Speech Sig. Processing* **ASSP-35**: 849–863. See also Corrections to "Real-valued fast Fourier transform algorithms" ^[13]
 - Peter D. Welch, 1969, A fixed-point fast Fourier transform error analysis ^[14], *IEEE Trans. Audio Electroacoustics* **17**: 151–157.
 - S. Winograd, 1978, On computing the discrete Fourier transform ^[15], *Math. Computation* **32**: 175–199.
-